



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών

«ΠΜΣ ΠΛΗΡΟΦΟΡΙΚΗ»

Μεταπτυχιακή Διατριβή

| | |
|-----------------------|---|
| Τίτλος Διατριβής | Ανάπτυξη Android εφαρμογής για την ανακύκλωση με ανάγνωση κωδικού barcode Implementation of an Android app for recycling by scanning a barcode |
| Όνοματεπώνυμο Φοιτητή | Μαρία Λυγερούδη |
| Πατρώνυμο | Παράσχος |
| Αριθμός Μητρώου | ΜΠΠΛ/ 18044 |
| Επιβλέπων | Ευθύμιος Αλέπης, Αναπληρωτής Καθηγητής |

Ημερομηνία Παράδοσης **Δεκέμβριος 2022**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Ευθύμιος Αλέπης
Αν. Καθηγητής

(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

(υπογραφή)

Κων/νος Πατσάκης
Αν. Καθηγητής



Περιεχόμενα

| | | |
|----------|--|----|
| 1. | Περίληψη | 8 |
| 2. | Εισαγωγή..... | 9 |
| 3. | Ανασκόπηση Πεδίου..... | 9 |
| 3.1. | Ανακύκλωση | 9 |
| 3.2. | Ελλάδα..... | 10 |
| 3.2.1. | Follow Green | 10 |
| 3.2.2. | The Green City | 11 |
| 3.2.3. | Ανακύκλωση Γυάλινων Συσκευασιών | 12 |
| 3.3. | Εξωτερικό | 13 |
| 3.3.1. | BC Recyclepedia | 14 |
| 3.3.2. | Recycle Smart | 16 |
| 3.3.3. | Trashly: <i>Recycling made easy</i> | 17 |
| 4. | Ανάλυση Απαιτήσεων..... | 19 |
| 4.2. | Λειτουργικές Απαιτήσεις..... | 19 |
| 5. | Παρουσίαση και Χρήση Εφαρμογής..... | 20 |
| 5.2. | Εγχειρίδιο Χρήστη | 20 |
| 5.1.1. | Οθόνη Σύνδεσης..... | 20 |
| 5.1.2. | Αρχική Οθόνη..... | 22 |
| 5.1.3. | Οθόνη προβολής συλλογής προϊόντων..... | 25 |
| 5.1.4. | Οθόνη προβολής στατιστικών | 28 |
| 5.1.5. | Προφίλ Χρήστη..... | 29 |
| 5.1.6. | Οθόνη αναγνώρισης κωδικού barcode..... | 31 |
| 5.2. | Υλοποίηση Εφαρμογής..... | 32 |
| 5.2.1. | Οθόνη Σύνδεσης..... | 33 |
| 5.2.1.1. | Παράθυρο Υπενθύμισης Κωδικού | 34 |
| 5.2.2. | Οθόνη Εγγραφής | 35 |
| 5.2.3. | Αρχική Οθόνη..... | 36 |
| 5.2.3.1. | Χωρίς Ανακυκλωμένα Προϊόντα | 36 |
| 5.2.3.2. | Παράθυρο Επιλογής Ημερομηνίας..... | 37 |
| 5.2.3.3. | Με Ανακυκλωμένα Προϊόντα | 38 |
| 5.2.3.4. | Παράθυρο Πληροφοριών για τα Μετάλλια | 39 |
| 5.2.4. | Οθόνη Προβολής Προϊόντων..... | 40 |
| 5.2.4.1. | Προεπιλογή | 40 |
| 5.2.4.2. | Παράθυρο Επιλογής Φίλτρου Προβολής | 41 |
| 5.2.5. | Οθόνη Προβολής Λεπτομερειών Προϊόντος..... | 42 |
| 5.2.6. | Οθόνη Προβολής Στατιστικών | 43 |
| 5.2.7. | Οθόνη Προφίλ..... | 44 |



| | |
|--|----|
| 5.2.7.1. Αλλαγή Άβαταρ | 45 |
| 5.2.7.2. Αλλαγή Κωδικού | 46 |
| 5.2.8. Οθόνη Σάρωσης Κωδικού Προϊόντος | 47 |
| 6. Σχεδιασμός | 47 |
| 6.2. Γλώσσες Προγραμματισμού | 47 |
| 6.2.1. Kotlin | 47 |
| 6.3. Εργαλεία | 48 |
| 6.3.1. Android Studio | 48 |
| 6.3.2. Firebase | 48 |
| 6.2.2.1. Firebase Crashlytics | 48 |
| 6.2.2.2. Firebase Authentication | 50 |
| 6.2.2.3. Firebase Realtime Database | 51 |
| 6.2.2.4. Firebase ML Kit | 54 |
| 6.3.3. Figma | 54 |
| 6.3.4. LottieFiles | 55 |
| 6.4. Βιβλιοθήκες | 56 |
| 6.4.1. Android Jetpack | 56 |
| 6.4.1.1. View Binding | 56 |
| 6.4.1.2. Fragments | 56 |
| 6.4.1.3. Dialog Fragments | 57 |
| 6.4.1.4. Navigation | 65 |
| 6.4.1.5. Room | 71 |
| 6.4.1.6. CameraX | 76 |
| 6.4.2. Material | 76 |
| 6.4.3. Google Fonts | 77 |
| 7. Συμπεράσματα και Μελλοντικές Επεκτάσεις | 78 |
| 7.1. Συμπεράσματα | 78 |
| 7.2. Μελλοντικές Επεκτάσεις | 78 |
| 8. Βιβλιογραφία | 79 |



Κατάλογος Εικόνων

| | |
|---|----|
| Εικόνα 1. Follow Green (Step 1) | 11 |
| Εικόνα 2. Follow Green (Step 2) | 11 |
| Εικόνα 3. Follow Green (Step3) | 11 |
| Εικόνα 4. Follow Green (Step 4) | 11 |
| Εικόνα 5. The Green City - Παρουσίαση ανακυκλώσιμων υλικών..... | 12 |
| Εικόνα 6. The Green City - Πρόγραμμα δρομολογίων..... | 12 |
| Εικόνα 7. The Green City - Χαρακτηριστικά..... | 12 |
| Εικόνα 8. Ανακύκλωση Γυάλινων Συσκευασιών - Αρχική Οθόνη | 13 |
| Εικόνα 9. Ανακύκλωση Γυάλινων Συσκευασιών - Προβολή κάδων σε ολόκληρη την Ελλάδα... 13 | |
| Εικόνα 10. Ανακύκλωση Γυάλινων Συσκευασιών - Προβολή κάδων σε μικρότερη περιοχή της Αττικής..... | 13 |
| Εικόνα 11. Ανακύκλωση Γυάλινων Συσκευασιών - Επιλογή κάδου..... | 13 |
| Εικόνα 12. Recyclepedia - Αρχική Οθόνη | 15 |
| Εικόνα 13. Recyclepedia - Κατηγορίες υλικών/ προϊόντων | 15 |
| Εικόνα 14. Recyclepedia - Υποκατηγορία υλικών/ προϊόντων | 15 |
| Εικόνα 15. Recyclepedia - Προβολή σημείων για ανακύκλωση επιλεγμένης υποκατηγορίας υλικού/ προϊόντος..... | 15 |
| Εικόνα 16. Recyclepedia - Προβολή σημείων ανακύκλωσης στον χάρτη | 16 |
| Εικόνα 17. Recyclepedia - Πληροφορίες για επιλεγμένο σημείο ανακύκλωσης | 16 |
| Εικόνα 18. RecycleSmart - Αρχική Οθόνη | 17 |
| Εικόνα 19. RecycleSmart - Επισκόπηση εφαρμογής..... | 17 |
| Εικόνα 20. RecycleSmart - Κατηγορίες υλικών/ προϊόντων..... | 17 |
| Εικόνα 21. RecycleSmart - Επιτεύγματα | 17 |
| Εικόνα 22. Trashly - Αρχική Οθόνη..... | 18 |
| Εικόνα 23. Trashly - Αναγνώριση ανακυκλώσιμου αντικειμένου με AI | 18 |
| Εικόνα 24. Trashly - Ανακάλυψη δυνατοτήτων ανακύκλωσης στην περιοχή του χρήστη | 19 |
| Εικόνα 25. Trashly - Ενημέρωση για την ανακυκλωσιμότητα αντικειμένων..... | 19 |
| Εικόνα 26. Trashly - Προβολή ανακυκλωμένων προϊόντων | 19 |
| Εικόνα 27. Trashly – Επιμορφωτικό Παιχνίδι..... | 19 |
| Εικόνα 28. Εγγραφή στην εφαρμογή..... | 21 |
| Εικόνα 29. Σύνδεση στην εφαρμογή | 21 |
| Εικόνα 30. Υπενθύμιση κωδικού σύνδεσης | 22 |
| Εικόνα 31. Παράδειγμα email επαναφοράς κωδικού | 22 |
| Εικόνα 32. Αρχική σελίδα χωρίς ανακυκλωμένα προϊόντα | 23 |
| Εικόνα 33. Αρχική σελίδα με προβολή ανακυκλωμένων προϊόντων | 23 |
| Εικόνα 34. Αναζήτηση προϊόντων ανά ημερομηνία | 23 |
| Εικόνα 35. Καταχώρηση ανακυκλωμένου προϊόντος | 24 |
| Εικόνα 36. Πληροφορίες Μεταλλίων Χρήστη | 24 |
| Εικόνα 37. Αναζήτηση προϊόντος..... | 25 |
| Εικόνα 38. Ενεργοποίηση φίλτρου προβολής ανακυκλώσιμων προϊόντων | 26 |
| Εικόνα 39. Ενεργοποίηση φίλτρου προβολής μη ανακυκλώσιμων προϊόντων | 26 |
| Εικόνα 40. Προβολή λεπτομερειών επιλεγμένου προϊόντος (Ανακυκλώσιμο)..... | 27 |
| Εικόνα 41. Προβολή λεπτομερειών επιλεγμένου προϊόντος (Μη ανακυκλώσιμο)..... | 28 |
| Εικόνα 42. Οθόνη στατιστικών (Χωρίς ύπαρξη δεδομένων)..... | 29 |
| Εικόνα 43. Οθόνη στατιστικών (Με ύπαρξη δεδομένων)..... | 29 |
| Εικόνα 44. Καταχώρηση username | 29 |
| Εικόνα 45. Αλλαγή εικόνας προφίλ | 30 |
| Εικόνα 46. Αλλαγή κωδικού σύνδεσης..... | 30 |
| Εικόνα 47. Αποσύνδεση από την εφαρμογή..... | 31 |
| Εικόνα 48. Επιτυχής αναγνώριση προϊόντος με barcode scanning..... | 31 |
| Εικόνα 49. Αποτυχία ταύτισης προϊόντος με τον αναγνωρισθέντα κωδικό barcode | 32 |



| | |
|---|----|
| Εικόνα 50. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Nexus S | 33 |
| Εικόνα 51. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Galaxy Nexus..... | 33 |
| Εικόνα 52. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Pixel 2XL..... | 33 |
| Εικόνα 53. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Pixel 4XL..... | 33 |
| Εικόνα 54. Στιγμιότυπο Παράθυρου Υπενθύμισης Κωδικού - Συσκευή Nexus S..... | 34 |
| Εικόνα 55. Στιγμιότυπο Παράθυρου Υπενθύμισης Κωδικού - Συσκευή Galaxy Nexus..... | 34 |
| Εικόνα 56. Στιγμιότυπο Παράθυρου Υπενθύμισης Κωδικού - Συσκευή Pixel 2XL..... | 34 |
| Εικόνα 57. Στιγμιότυπο Παράθυρου Υπενθύμισης Κωδικού - Συσκευή Pixel 4XL..... | 34 |
| Εικόνα 58. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Nexus S..... | 35 |
| Εικόνα 59. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Galaxy Nexus..... | 35 |
| Εικόνα 60. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Pixel 2XL..... | 35 |
| Εικόνα 61. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Pixel 4XL..... | 35 |
| Εικόνα 62. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Nexus S..... | 36 |
| Εικόνα 63. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Galaxy Nexus..... | 36 |
| Εικόνα 64. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Pixel 2XL..... | 36 |
| Εικόνα 65. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Pixel 4XL..... | 36 |
| Εικόνα 66. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Nexus S..... | 37 |
| Εικόνα 67. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Galaxy Nexus..... | 37 |
| Εικόνα 68. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Pixel 2XL..... | 37 |
| Εικόνα 69. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Pixel 4XL..... | 37 |
| Εικόνα 70. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Nexus S..... | 38 |
| Εικόνα 71. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Galaxy Nexus..... | 38 |
| Εικόνα 72. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Pixel 2XL..... | 38 |
| Εικόνα 73. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Pixel 4XL..... | 38 |
| Εικόνα 74. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Nexus S..... | 39 |
| Εικόνα 75. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Galaxy Nexus..... | 39 |
| Εικόνα 76. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Pixel 2XL..... | 39 |
| Εικόνα 77. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Pixel 4XL..... | 39 |
| Εικόνα 78. Στιγμιότυπο Οθόνης Προβολής Προϊόντων - Συσκευή Nexus S..... | 40 |
| Εικόνα 79. Στιγμιότυπο Οθόνης Προβολής Προϊόντων - Συσκευή Galaxy Nexus..... | 40 |
| Εικόνα 80. Στιγμιότυπο Οθόνης Προβολής Προϊόντων - Συσκευή Pixel 2XL..... | 40 |
| Εικόνα 81. Στιγμιότυπο Οθόνης Προβολής Προϊόντων - Συσκευή Pixel 4XL..... | 40 |
| Εικόνα 82. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής - Συσκευή Nexus S..... | 41 |
| Εικόνα 83. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής - Συσκευή Galaxy Nexus..... | 41 |
| Εικόνα 84. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής Προϊόντων - Συσκευή Pixel 2XL..... | 41 |
| Εικόνα 85. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής - Συσκευή Pixel 4XL..... | 41 |
| Εικόνα 86. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Nexus S..... | 42 |
| Εικόνα 87. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Galaxy Nexus..... | 42 |
| Εικόνα 88. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Pixel 2XL..... | 42 |
| Εικόνα 89. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Pixel 4XL..... | 42 |
| Εικόνα 90. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Nexus S..... | 43 |
| Εικόνα 91. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Galaxy Nexus..... | 43 |
| Εικόνα 92. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Pixel 2XL..... | 43 |
| Εικόνα 93. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Pixel 4XL..... | 43 |
| Εικόνα 94. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Nexus S..... | 44 |
| Εικόνα 95. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Galaxy Nexus..... | 44 |
| Εικόνα 96. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Pixel 2XL..... | 44 |
| Εικόνα 97. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Pixel 4XL..... | 44 |
| Εικόνα 98. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Nexus S..... | 45 |
| Εικόνα 99. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Galaxy Nexus..... | 45 |
| Εικόνα 100. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Pixel 2XL..... | 45 |



| | |
|--|----|
| Εικόνα 101. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Pixel 4XL..... | 45 |
| Εικόνα 102. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Nexus S..... | 46 |
| Εικόνα 103. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Galaxy Nexus..... | 46 |
| Εικόνα 104. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Pixel 2XL..... | 46 |
| Εικόνα 105. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Pixel 4XL..... | 46 |
| Εικόνα 106. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος - Συσκευή Nexus S..... | 47 |
| Εικόνα 107. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος - Συσκευή Galaxy Nexus..... | 47 |
| Εικόνα 108. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος - Συσκευή Pixel 2XL..... | 47 |
| Εικόνα 109. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος..... | 47 |
| Εικόνα 110. Αρχική οθόνη Crashlytics..... | 49 |
| Εικόνα 111. Επισκόπηση σφάλματος..... | 50 |
| Εικόνα 112. Επισκόπηση στοιχείων σχετικών με το υπό επεξεργασία σφάλμα..... | 50 |
| Εικόνα 113. Αυθεντικοποίηση χρηστών..... | 51 |
| Εικόνα 114. Realtime Database - Βασικοί κόμβοι..... | 51 |
| Εικόνα 115. Realtime Database - Κόμβος Προϊόντων..... | 52 |
| Εικόνα 116. Realtime Database - Κόμβος Συγκεκριμένου Προϊόντος..... | 52 |
| Εικόνα 117. Realtime Database - Κόμβος χρήστη..... | 53 |
| Εικόνα 118. Realtime Database – Κόμβος Ανακυκλωμένων Προϊόντων..... | 54 |
| Εικόνα 119. Στιγμιότυπο οθόνης από το Figma..... | 55 |
| Εικόνα 121. Προσθήκη dependency στο project για τη χρήση Lottie αρχείων..... | 55 |
| Εικόνα 122. Στιγμιότυπο χρήσης Lottie αρχείο στην εφαρμογή..... | 56 |
| Εικόνα 123. Πρότυπο αναδυομένου παραθύρου για την επιβεβαίωση ενέργειας..... | 57 |
| Εικόνα 124. Κώδικας XML για τη δημιουργία ειδικού layout αναδυομένου παραθύρου - Στιγμιότυπο Οθόνης 1..... | 58 |
| Εικόνα 125. Κώδικας XML για τη δημιουργία ειδικού layout αναδυομένου παραθύρου - Στιγμιότυπο Οθόνης 2..... | 59 |
| Εικόνα 126. Κώδικας XML για τη δημιουργία ειδικού layout αναδυομένου παραθύρου - Στιγμιότυπο Οθόνης 3..... | 60 |
| Εικόνα 127. Κώδικας για την υλοποίηση αναδυόμενου παραθύρου με τη χρήση του Dialog Fragment - Στιγμιότυπο Οθόνης 1..... | 62 |
| Εικόνα 128. Κώδικας για την υλοποίηση αναδυόμενου παραθύρου με τη χρήση του Dialog Fragment - Στιγμιότυπο Οθόνης 2..... | 63 |
| Εικόνα 129. Κώδικας για την υλοποίηση αναδυόμενου παραθύρου με τη χρήση του Dialog Fragment - Στιγμιότυπο Οθόνης 3..... | 64 |
| Εικόνα 130. Στιγμιότυπο οθόνης για την κλήση αναδυόμενου παραθύρου..... | 64 |
| Εικόνα 131. Στιγμιότυπο οθόνης - Ορισμός δεδομένου προς μεταφορά κατά την πλοήγηση μεταξύ προορισμών..... | 65 |
| Εικόνα 132. Στιγμιότυπο οθόνης 1 - Αρχείο Navigation Graph..... | 66 |
| Εικόνα 133. Στιγμιότυπο οθόνης 2 - Αρχείο Navigation Graph..... | 67 |
| Εικόνα 134. Στιγμιότυπο οθόνης 3 - Αρχείο Navigation Graph..... | 68 |
| Εικόνα 135. Οπτική παρουσίαση του Navigation Graph - Στιγμιότυπο οθόνης..... | 69 |
| Εικόνα 136. Ενσωμάτωση NavHost σε μία Activity - Στιγμιότυπο οθόνης..... | 70 |
| Εικόνα 137. Πρόσβαση στον NavHost – Στιγμιότυπο οθόνης..... | 70 |
| Εικόνα 138. Ορισμός NavController - Στιγμιότυπο οθόνης..... | 70 |
| Εικόνα 139. Τμήμα κώδικα για την πλοήγηση μεταξύ προορισμών με τη χρήση του NavController – Στιγμιότυπο οθόνης 1..... | 70 |
| Εικόνα 140. Τμήμα κώδικα για την πλοήγηση μεταξύ προορισμών με τη χρήση του NavController, μεταφέροντας δεδομένα μεταξύ τους – Στιγμιότυπο οθόνης 2..... | 71 |
| Εικόνα 141. Room - Database Class..... | 72 |
| Εικόνα 142. Room - Product Entity..... | 73 |
| Εικόνα 143. Room - BaseDao..... | 74 |
| Εικόνα 144. Room - Recycled Product Dao..... | 74 |
| Εικόνα 145. Διάγραμμα Τάξης UML..... | 75 |



Εικόνα 146. Αρχιτεκτονική της Room 76



1. Περίληψη

Η παρούσα μεταπτυχιακή διατριβή αφορά στην ανάπτυξη μίας εφαρμογής για Android συσκευές, η οποία παρέχει ανατροφοδότηση σχετικά με την ανακυκλωσιμότητα ή μη ενός προϊόντος. Η ανατροφοδότηση έχει τη μορφή οδηγιών ανακύκλωσης, ή μη σε αντίθετη περίπτωση. Για την λήψη αυτών, ο χρήστης καλείται να σαρώσει τον κωδικό ενός προϊόντος, να αναζητήσει βάσει του ονόματός του, ή να ανατρέξει μία λίστα προϊόντων που προβάλλονται στην εφαρμογή.

Η εφαρμογή αναπτύχθηκε σε περιβάλλον Android, στη γλώσσα προγραμματισμού Kotlin. Για τον σχεδιασμό της, πριν ακόμα λάβει μορφή, επιλέχθηκε η χρήση του εργαλείου Figma, ενώ για την υλοποίησή της χρησιμοποιήθηκε το IDE Android Studio, σε συνδυασμό με πλήθος εργαλείων και βιβλιοθηκών. Άξιο αναφοράς είναι το Barcode Scanning (ML Kit) της Firebase, το οποίο κατέστησε δυνατή την ανάγνωση κωδικών barcode μέσω της κάμερας του κινητού.

Abstract

The current postgraduate dissertation consists of the development of an app for Android devices, that provides feedback regarding the recyclability or not of a certain product. This feedback consists of instructions for recycling, or not in the opposite case. To receive that, the user needs to scan the barcode of a product, search by its name, or browse through a list of products displayed in the app.

The app was developed in the Android environment, in the Kotlin programming language. For the designs of the app the Figma tool was used, while for the implementation IDE Android Studio was used, in combination with a selection of tools and libraries. A n honorable mention is the Barcode Scanning (ML Kit) from Firebase that made the scanning of barcodes possible through the mobile phone's camera.



2. Εισαγωγή

Παρά την αύξηση που έχει παρατηρηθεί στο εγχείρημα της ανακύκλωσης, συνεχίζει η αποτελεσματικότητά της να είναι μειωμένη. Πολλά από τα άτομα που ανακυκλώνουν, παρά τη θέληση που επιδεικνύουν, κάνουν λάθη. Τα λάθη αυτά, ακόμα κι αν φαίνονται μικρά ή μεμονωμένα μπορεί να έχουν μοιραίο αποτέλεσμα για το περιεχόμενο ενός ολόκληρου κάδου, επιμολύνοντάς τον στο σύνολό του. Ένα μόνο μπουκάλι από γάλα, που δεν έχει ξεπλυθεί σωστά, μπορεί να καταλήξει να αχρηστεύσει και τον περιγύρο του.

Μία έρευνα¹ από τις επιχειρήσεις ISRI (Scrap Recycling Industries) που διεξήχθη το 2018 υπέδειξε ότι το 66% των Αμερικανών δεν θα ανακυκλώσουν, εάν δεν είναι εύκολο να το κάνουν. Οι λόγοι για τους οποίους συμβαίνει αυτό είναι αναμφίβολα πολλοί και διαφορετικοί, λ.χ. προσβασιμότητα, ή το ενδιαφέρον για το περιβάλλον. Εμείς όμως επιλέγουμε να εστιάσουμε σε έναν: η επιμόρφωση σχετικά με την ανακύκλωση, και κατ' επέκταση η εύκολη πρόσβαση σε αυτή. Είναι γνωστό ότι παρά τη σημασία που έχει η ανακύκλωση, οι άνθρωποι δεν είναι σωστά ή και καθόλου ενημερωμένοι για την ίδια, αλλά και τη σημασία της. Από την άλλη, εκείνοι που έχουν γνώσεις αναφορικά με την ανακύκλωση, δεν είναι απαραίτητο ότι συγκεντρώνουν τις σωστές από αυτές. Δεν είναι εύκολο εξάλλου, δεδομένου ότι η διεξαγωγή της ανακύκλωσης διαφέρει ανάλογα με τον τόπο. Λαμβάνοντας αυτά υπόψιν, καταλήξαμε στο συμπέρασμα πως οποιαδήποτε πρωτοβουλία ως προς την εύκολη ενημέρωση των πολιτών, μόνο καλό μπορεί να επιφέρει στην κοινωνία. Έτσι λοιπόν γεννήθηκε η ιδέα της παρούσας εφαρμογής, η οποία παρέχει ενημερωτικό υλικό για την ανακυκλωσιμότητα των προϊόντων, καθώς και τις πρακτικές σωστής ανακύκλωσης για το κάθε προϊόν. Ο χρήστης μπορεί εύκολα να ενημερωθεί για ένα προϊόν με τη χρήση και μόνο του κινητού του τηλεφώνου. Σαρώνοντας τον κωδικό barcode του προϊόντος ή αναζητώντας το όνομά του, είναι δυνατόν να ανακαλύψει πληροφορίες σχετικά με την ανακύκλωσή του που μπορεί να του φανούν χρήσιμες.

3. Ανασκόπηση Πεδίου

3.1. Ανακύκλωση

Ανακύκλωση ονομάζεται η διαδικασία επανεπεξεργασίας των υλικών κατά το τέλος του κύκλου ζωής τους επιστρέφοντάς τα με αυτόν τον τρόπο ξανά στην εφοδιαστική αλυσίδα.² Πλέον, βέβαια, ο όρος της ανακύκλωσης έχει επεκταθεί, καθιστώντας τη συνυφασμένη με το τρίπτυχο των *Rs*, τα οποία αντιπροσωπεύουν ουσιαστικές έννοιες για τη μείωση του περιβαλλοντικού αντίκτυπου των ανθρώπων στον πλανήτη. Χωρίς τις άλλες δύο έννοιες, η ανακύκλωση καθίσταται ελλιπής. Πιο συγκεκριμένα, το τρίπτυχο αυτό περιλαμβάνει την Ελάττωση (*Reduce*), την Επαναχρησιμοποίηση (*Reuse*), και τέλος, την Ανακύκλωση (*Recycle*). Ο συνδυασμός τους βοηθά την ανακύκλωση ελαφραίνοντάς τη από το επιπλέον βάρος, ενώ παράλληλα, την καθιστά πιο αποτελεσματική.³

Η σημασία της ανακύκλωσης συνίσταται σε ένα πλήθος παραγόντων που ενισχύουν την ανάγκη να την επιδιώξουμε ως κοινωνία. Πρώτο παράγοντα αποτελεί η εξοικονόμηση πρώτων υλών. Με αυτήν, τα υλικά μετατρέπονται σε νέα προϊόντα και συνεπώς μειώνεται η ανάγκη για χρήση φυσικών πόρων, γεγονός που συμβάλλει στην προστασία των φυσικών περιβαλλόντων για το μέλλον. Επιπλέον, η ενέργεια που χρειάζεται για την επεξεργασία των προϊόντων προς ανακύκλωση είναι αισθητά λιγότερη από εκείνη που θα χρειαστεί για την εκ νέου παραγωγή προϊόντων από πρώτες ύλες, επομένως μειώνεται η εκπομπή αερίων του θερμοκηπίου και υποβοηθάται η αντιμετώπιση της κλιματικής αλλαγής. Πέραν της ενέργειας που εξοικονομείται, περιορίζεται και η ρύπανση των υδάτων και αέρα, καθώς ελαττώνεται η ανάγκη εξαγωγής, διύλισης και επεξεργασίας πρώτων υλών. Τέλος, χάρη στην επαναχρησιμοποίηση υλικών, μειώνεται ο όγκος των σκουπιδιών που μεταφέρονται στους σκουπιδοτοπούς, άρα και η ανάγκη διατήρησής τους, ή εύρεσης νέων.⁴



3.2. Ελλάδα

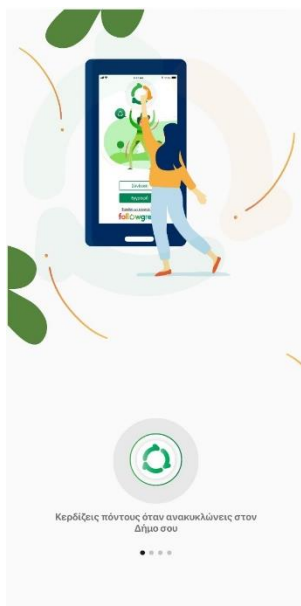
Παρόλη τη σημασία της ανακύκλωσης, δυστυχώς δεν υπάρχει αρκετό υλικό σχετικά με την ανακύκλωση στην Ελλάδα. Οι εφαρμογές που βρέθηκαν κατά την αναζήτηση ήταν τρεις συνολικά σε πλήθος. Παρουσιάζονται αναλυτικά παρακάτω.

3.2.1. Follow Green

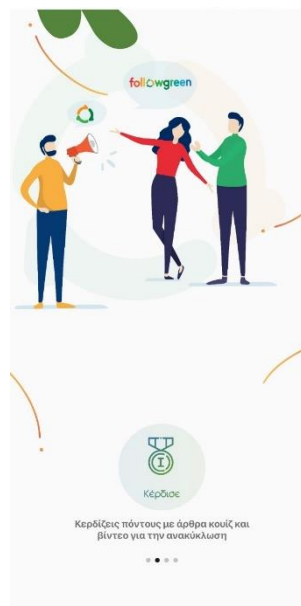
Πρόκειται για μία εξαιρετική πρωτοβουλία, όπου ο χρήστης επιβραβεύεται με πόντους μαθαίνοντας για την ανακύκλωση μέσω άρθρων και ανακυκλώνοντας. Οι πόντοι αυτοί μπορούν να εξαργυρωθούν σε συνεργαζόμενα καταστήματα ή να δωρισθούν σε σχολεία που ανήκουν στον Δήμο του χρήστη. Η εφαρμογή είναι διαθέσιμη ως web application⁵, αλλά και ως mobile εφαρμογή στο κινητό⁶.

Παρακάτω προβάλλονται οι δυνατότητες που παρέχονται στον χρήστη της εφαρμογής:

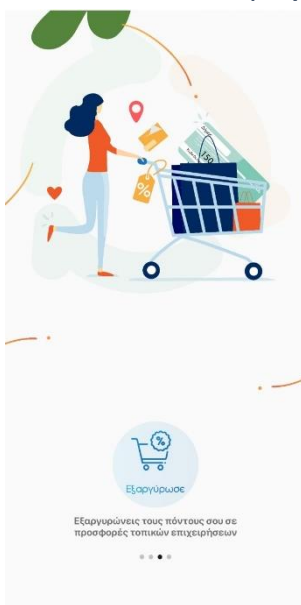




Εικόνα 1. Follow Green (Step 1)



Εικόνα 2. Follow Green (Step 2)



Εικόνα 3. Follow Green (Step3)



Εικόνα 4. Follow Green (Step 4)

Ένα μειονέκτημα της εφαρμογής είναι ότι αφορά συγκεκριμένους μόνο Δήμους της Ελλάδας, επομένως καθιστά αδύνατη τη συμμετοχή χρηστών που δεν ανήκουν σε αυτούς. Ελπίζουμε στο μέλλον να ενισχυθεί η προσπάθεια αυτή ώστε να συμμετέχει το σύνολο των Δήμων.

3.2.2. The Green City

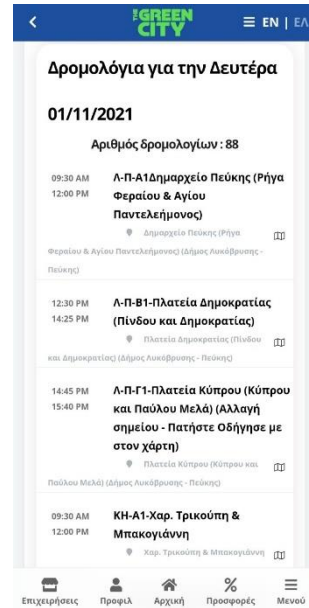
Το παρόν πρόγραμμα αφορά στο σύνολο των Δήμων του νομού της Αττικής. Σκοπός είναι η μείωση του περιβαλλοντικού αποτυπώματος των εμπλεκόμενων Δήμων και η συνεργασία με τους πολίτες για μία καλύτερη προσέγγιση της ανακύκλωσης. Δύο φορές την εβδομάδα προγραμματίζονται ειδικά δρομολόγια συλλογής ανακυκλώσιμων αντικειμένων με τα «ΚΙΝΗΤΑ ΠΡΑΣΙΝΑ ΣΗΜΕΙΑ». Οι χρήστες μπορούν να ενημερωθούν για αυτά ανάλογα με τον Δήμο στον οποίο ανήκουν στο: <https://thegreencity.gr/el/routes>.



Οι χρήστες μπορούν να προμηθευτούν την εφαρμογή μέσω του Google PlayStore ή στον ιστότοπο: <https://thegreencity.gr/>. Μερικά στιγμιότυπα της εφαρμογής παρουσιάζονται παρακάτω:



Εικόνα 5. The Green City - Παρουσίαση ανακυκλώσιμων υλικών



Εικόνα 6. The Green City - Πρόγραμμα δρομολογίων



Εικόνα 7. The Green City - Χαρακτηριστικά

Όπως και στην προηγούμενη εφαρμογή, έτσι και σε αυτήν δυστυχώς, δεν υποστηρίζεται το σύνολο των Δήμων, δυσκολεύοντας έτσι τους χρήστες που επιθυμούν να συμμετέχουν.

3.2.3. Ανακύκλωση Γυάλινων Συσκευασιών

Η εφαρμογή αυτή σκοπό έχει την παράθεση πληροφοριών σχετικά με την τοποθεσία κάδων για την ανακύκλωση γυάλινων συσκευασιών⁷. Στον αντίστοιχο ιστότοπο παρέχονται και ενημερωτικές πληροφορίες σχετικά με τα υλικά που είναι δυνατόν να τοποθετηθούν στους κάδους αυτούς. Παρακάτω παρουσιάζονται στιγμιότυπα από την εφαρμογή:

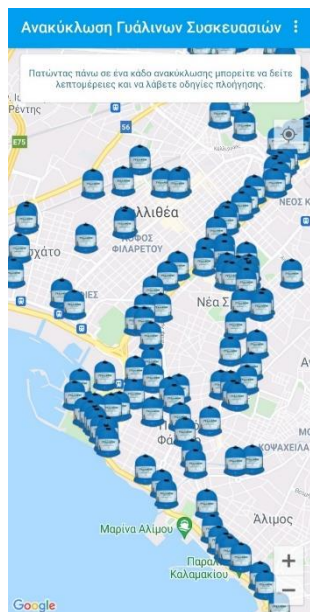




Εικόνα 8. Ανακύκλωση Γυάλινων Συσκευασιών - Αρχική Οθόνη



Εικόνα 9. Ανακύκλωση Γυάλινων Συσκευασιών - Προβολή κάδων σε ολόκληρη την Ελλάδα



Εικόνα 10. Ανακύκλωση Γυάλινων Συσκευασιών - Προβολή κάδων σε μικρότερη περιοχή της Αττικής



Εικόνα 11. Ανακύκλωση Γυάλινων Συσκευασιών - Επιλογή κάδου

3.3. Εξωτερικό

Οι εφαρμογές που συναντώνται σε παγκόσμιο επίπεδο είναι περισσότερες και συχνά περιέχουν εκτενέστερο πλήθος πληροφοριών, συνδυάζοντας τη γνώση με δημιουργικούς τρόπους για την κινητοποίηση ή διευκόλυνση των πολιτών. Δυστυχώς, καμία από αυτές δεν έχει παγκόσμια εμβέλεια, ώστε να μπορεί να χρησιμοποιηθεί από όλα τα μέρη του πλανήτη. Παρακάτω παρουσιάζονται μερικές από αυτές.



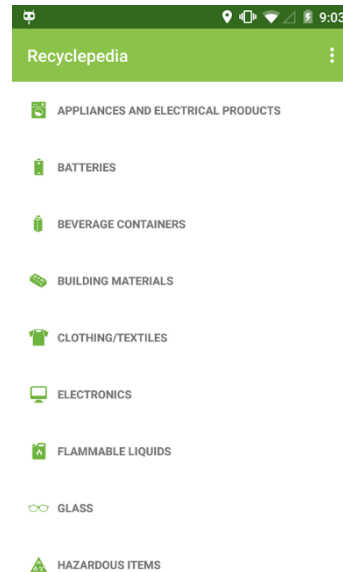
3.3.1. BC Recyclepedia

Η εφαρμογή BC Recyclepedia⁸ έχει έδρα την Βρετανική Κολομβία (Καναδάς) και περιλαμβάνει μία λίστα με δυνατότητες ανακύκλωσης για πάνω από 70 αντικείμενα. Με την επιλογή μίας κατηγορίας υλικών/ προϊόντων, η εφαρμογή παρέχει πληροφορίες για τα διαθέσιμα σημεία ανακύκλωσης, όπως τοποθεσία, ώρες λειτουργίας, στοιχεία επικοινωνίας. Παράδειγμα χρήσης της εφαρμογής φαίνεται παρακάτω:

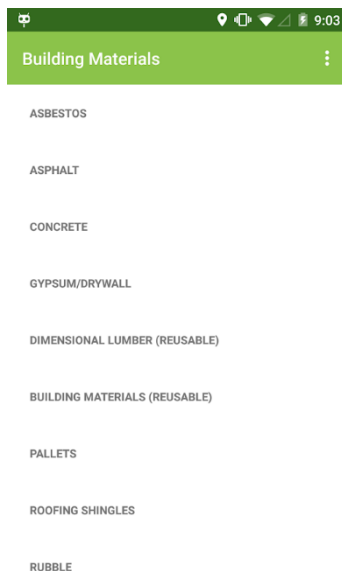




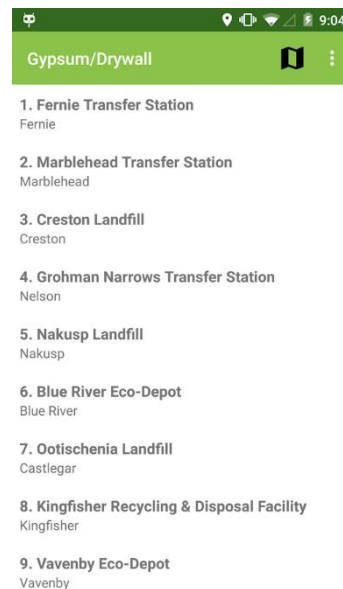
Εικόνα 12. Recyclepedia - Αρχική Οθόνη



Εικόνα 13. Recyclepedia - Κατηγορίες υλικών/ προϊόντων

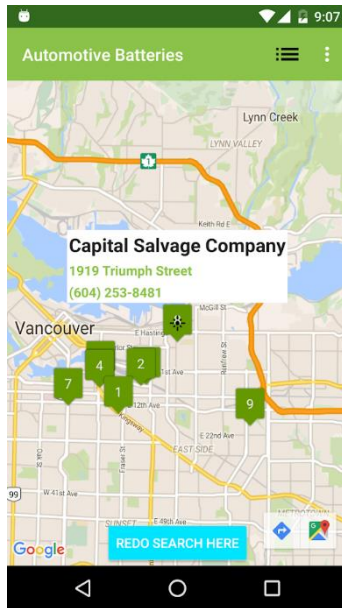


Εικόνα 14. Recyclepedia - Υποκατηγορία υλικών/ προϊόντων

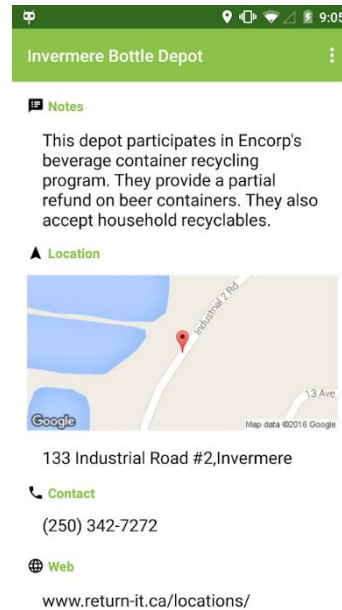


Εικόνα 15. Recyclepedia - Προβολή σημείων για ανακύκλωση επιλεγμένης υποκατηγορίας υλικού/ προϊόντος





Εικόνα 16. Recyclepedia - Προβολή σημείων ανακύκλωσης στον χάρτη

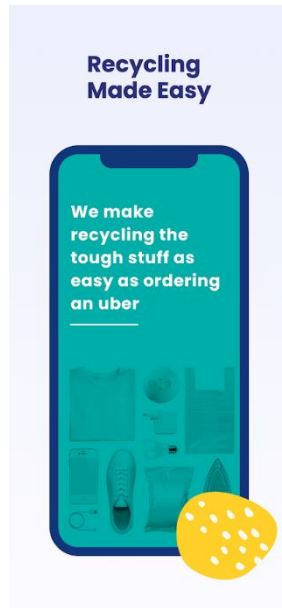


Εικόνα 17. Recyclepedia - Πληροφορίες για επιλεγμένο σημείο ανακύκλωσης

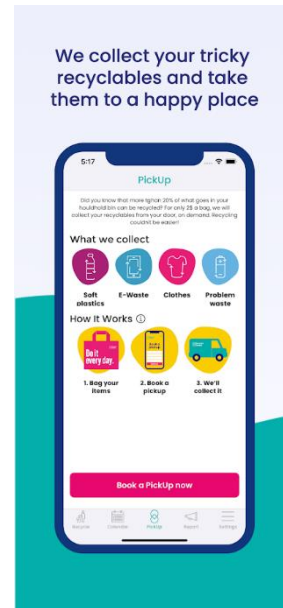
3.3.2. Recycle Smart

Η εφαρμογή RecycleSmart⁹ (Αυστραλία) υποστηρίζει την ανακύκλωση παραπάνω από διακόσια αντικείμενα. Παρέχει πολύτιμες πληροφορίες για την ανακύκλωσή τους, με σκοπό την καλύτερη ενημέρωση των πολιτών, αλλά και τη διευκόλυνσή τους στη διαδικασία της. Συγκεκριμένα, ο χρήστης επιλέγει μία ημερομηνία που τον εξυπηρετεί και με αυτόν τον τρόπο κανονίζεται η συλλογή των ανακυκλώσιμων αγαθών του, χωρίς καν να είναι απαραίτητη η παρουσία του. Ο διαχωρισμός των αντικειμένων γίνεται από εξειδικευμένα άτομα μετά τη συλλογή τους. Παρακάτω προβάλλονται στιγμιότυπα της εφαρμογής¹⁰.

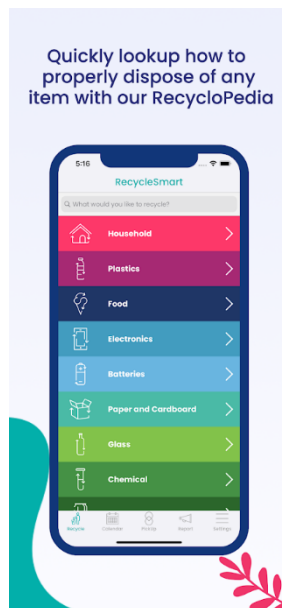




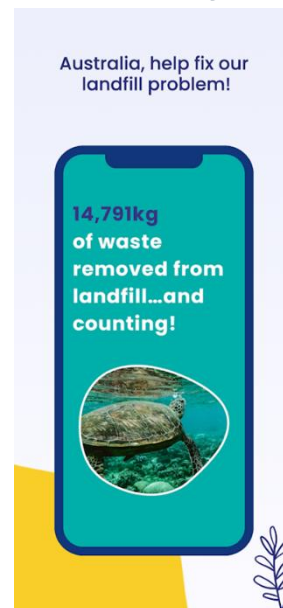
Εικόνα 18. RecycleSmart - Αρχική Οθόνη



Εικόνα 19. RecycleSmart - Επισκόπηση εφαρμογής



Εικόνα 20. RecycleSmart - Κατηγορίες υλικών/ προϊόντων



Εικόνα 21. RecycleSmart - Επιτεύγματα

3.3.3. Trashly: *Recycling made easy*

Το Trashly¹¹ είναι μία εφαρμογή με έδρα την Αμερική, που στόχο έχει τη διευκόλυνση της ανακύκλωσης με την παροχή πλήθους μεθόδων:

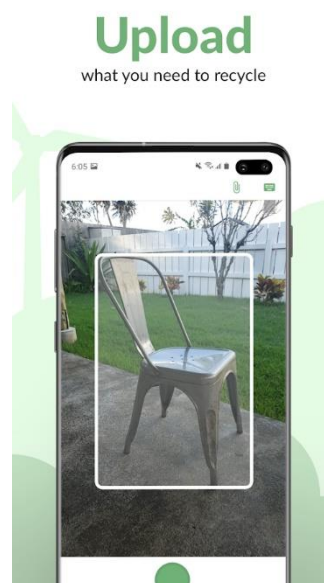
- Χρήση τεχνητής νοημοσύνης για τη διευκόλυνση στην αναγνώριση ανακυκλώσιμων ή μη προϊόντων. Ο χρήστης βγάζει φωτογραφία το προϊόν που επιθυμεί να ανακυκλώσει και η εφαρμογή παρέχει ανατροφοδότηση αναφορικά με το αν είναι ανακυκλώσιμο ή μη.
- Παροχή πληροφοριών για τις δυνατότητες ανακύκλωσης στην περιοχή του χρήστη.
- Προβολή του συνόλου των ανακυκλωμένων προϊόντων που έχει καταχωρήσει ο χρήστης.
- Ένα ενημερωτικού περιεχομένου παιχνίδι κατά το οποίο ο χρήστης καλείται να κατατάξει τα αντικείμενα που πέφτουν από τον ουρανό στον κατάλληλο κάδο. Οι διαθέσιμοι κάδοι είναι οι:



Πλαστικού, Γυαλιού, Ηλεκτρονικών Συσκευών, Βιολογικού Κινδύνου και Οργανικών Απόβλητων.



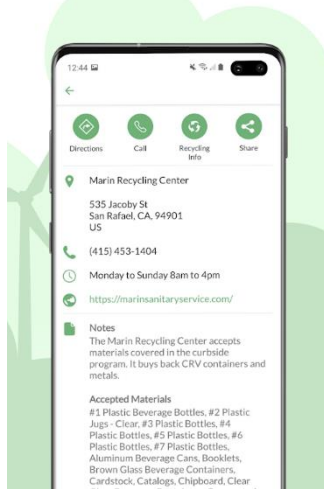
Εικόνα 22. Trashly - Αρχική Οθόνη



Εικόνα 23. Trashly - Αναγνώριση ανακυκλώσιμου αντικειμένου με AI

Explore

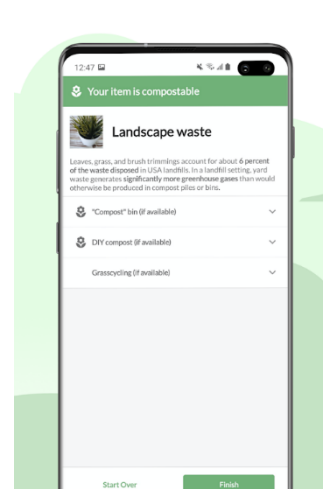
recycling options near you



Εικόνα 24. Trashly - Ανακάλυψη δυνατοτήτων ανακύκλωσης στην περιοχή του χρήστη

Learn

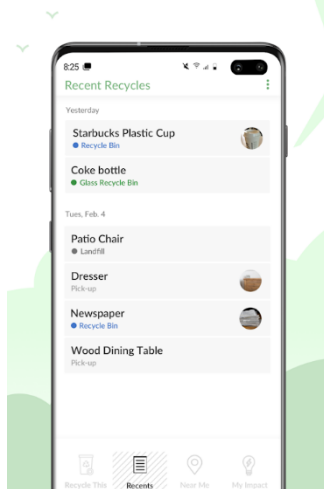
how to help the environment



Εικόνα 25. Trashly - Ενημέρωση για την ανακυκλωσιμότητα αντικειμένων

Track

your recycling efforts



Εικόνα 26. Trashly - Προβολή ανακυκλωμένων προϊόντων

Play

Catch-The-Trash



Εικόνα 27. Trashly – Επιμορφωτικό Παιχνίδι

4. Ανάλυση Απαιτήσεων

4.2. Λειτουργικές Απαιτήσεις

- Ο χρήστης πρέπει να εγγραφεί για να εισέλθει στην εφαρμογή, εισάγοντας τη διεύθυνση ηλεκτρονικού ταχυδρομείου του και έναν κωδικό πρόσβασης τύπου αλφαριθμητικού, τουλάχιστον έξι χαρακτήρων.
- Ο χρήστης θα πρέπει να συνδεθεί για να εισέλθει στην εφαρμογή, χρησιμοποιώντας τα στοιχεία που έχει θέσει κατά την εγγραφή του (διεύθυνση ηλεκτρονικού ταχυδρομείου και κωδικός πρόσβασης).



- Ο χρήστης θα πρέπει να έχει τη δυνατότητα υπενθύμισης του κωδικού πρόσβασής του, ώστε να μην αποκλειστεί από την εφαρμογή στην περίπτωση που τον ξεχάσει.
- Ο χρήστης θα πρέπει να έχει τη δυνατότητα να αλλάξει τον κωδικό πρόσβασής του, αν το επιθυμήσει.
- Πρέπει να επιτρέπεται στον χρήστη η δυνατότητα αποσύνδεσης από την εφαρμογή. Πριν ολοκληρωθεί η ενέργεια αυτή, πρέπει να υπάρχει επιβεβαίωση από τον χρήστη.
- Ο χρήστης πρέπει να μπορεί να ανατρέξει τη λίστα προϊόντων που βρίσκονται στη βάση, να ελέγξει ποιο από αυτά είναι ανακυκλώσιμο ή όχι και να προβεί στην ανάγνωση των πληροφοριών σχετικά με την ανακύκλωσή του ή μη.
- Ο χρήστης πρέπει να μπορεί να προβάλει τα προϊόντα με βάση το κριτήριο της ανακυκλωσιμότητάς τους ή μη.
- Ο χρήστης πρέπει να μπορεί να αναζητήσει ένα προϊόν βάσει του ονόματός του. Αν αυτό βρίσκεται στη βάση δεδομένων, τότε προβάλλονται οι πληροφορίες του.
- Ο χρήστης πρέπει να μπορεί να δηλώσει ότι έχει ανακυκλώσει κάποιο προϊόν. Αυτό θα καταχωρείται στη βάση δεδομένων με βάση του ειδικό του αναγνωριστικό και το timestamp της στιγμής καταχώρησης.
- Ο χρήστης πρέπει να μπορεί να ανατρέξει μία λίστα των προϊόντων που έχει ανακυκλώσει ως τώρα με βάση την ημερομηνία ανακύκλωσης.
- Ο χρήστης πρέπει να ελέγχει την πρόοδο του όσον αφορά στην προσωπική του ανακύκλωση προϊόντων με ένα σύστημα ποντοδότησης. Το σύστημα θα έχει τέσσερις βαθμίδες, εκ των οποίων η πρώτη συναντάται στα δέκα ανακυκλωμένα προϊόντα, η δεύτερη στα πενήντα, η τρίτη στα εκατόν πενήντα και η τελευταία στα τριακόσια.
- Πρέπει να επιτρέπεται στον χρήστη να ορίσει και να αλλάξει το όνομα χρήστη του.
- Πρέπει να επιτρέπεται στον χρήστη να αλλάξει το άβατάρ του, μέσω μίας λίστας τοπικά αποθηκευμένων εικόνων.
- Ο χρήστης πρέπει να μπορεί να ανατρέξει μία λίστα των πιο συχνά ανακυκλωμένων προϊόντων βάσει δεδομένων που συλλέχθηκαν από το σύνολο των χρηστών της εφαρμογής.
- Ο χρήστης πρέπει να μπορεί να μάθει ποιο είναι το πιο συχνά ανακυκλωμένο υλικό βάσει δεδομένων που συλλέχθηκαν από το σύνολο των χρηστών της εφαρμογής.
- Ο χρήστης πρέπει να μπορεί να μάθει ποιο είναι τον συνολικό αριθμό ανακυκλωμένων προϊόντων βάσει δεδομένων που συλλέχθηκαν από το σύνολο των χρηστών της εφαρμογής.
- Πρέπει να επιτρέπεται στον χρήστη να σαρώνει μέσω της κάμεράς του κάποιον κωδικό barcode προϊόντος, προκειμένου να ελέγξει εάν αυτό είναι ανακυκλώσιμο ή όχι. Το σύστημα να επιστρέψει στον χρήστη τις πληροφορίες σχετικά το προϊόν προς σάρωση, ή σε περίπτωση που δεν την κατέχει να προβάλλει την αντίστοιχη ανατροφοδότηση.

5. Παρουσίαση και Χρήση Εφαρμογής

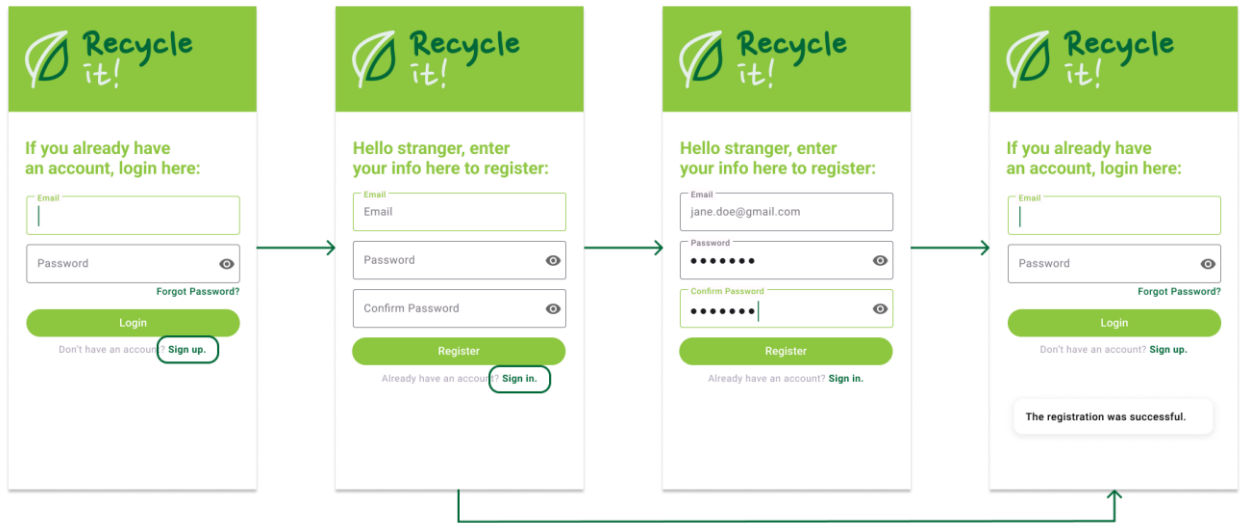
5.2. Εγχειρίδιο Χρήστη

5.1.1. Οθόνη Σύνδεσης

Κατά το άνοιγμα της εφαρμογής, προβάλλεται στον χρήστη η οθόνη *Σύνδεσης*, από όπου έχει τις ακόλουθες δυνατότητες:

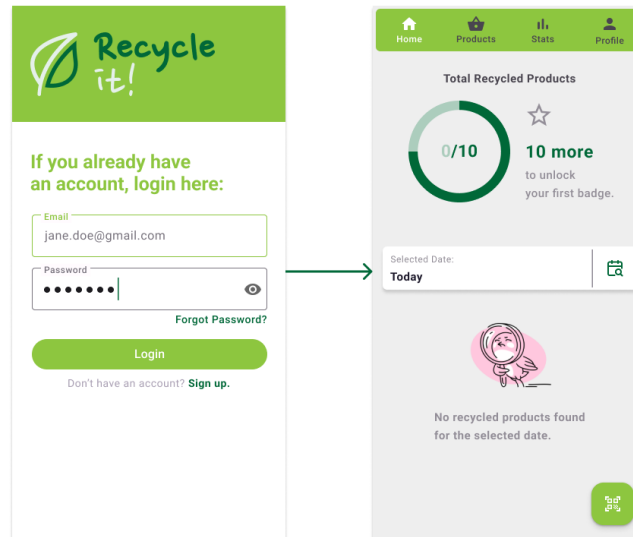
- Εγγραφή, αν δεν έχει ήδη λογαριασμό.





Εικόνα 28. Εγγραφή στην εφαρμογή

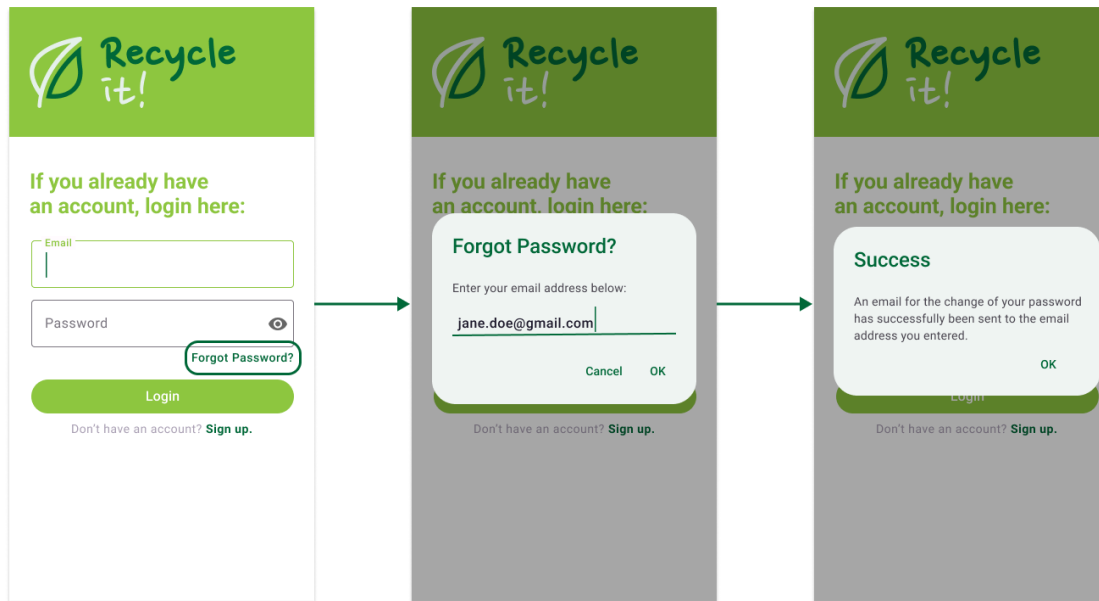
- Σύνδεση.



Εικόνα 29. Σύνδεση στην εφαρμογή

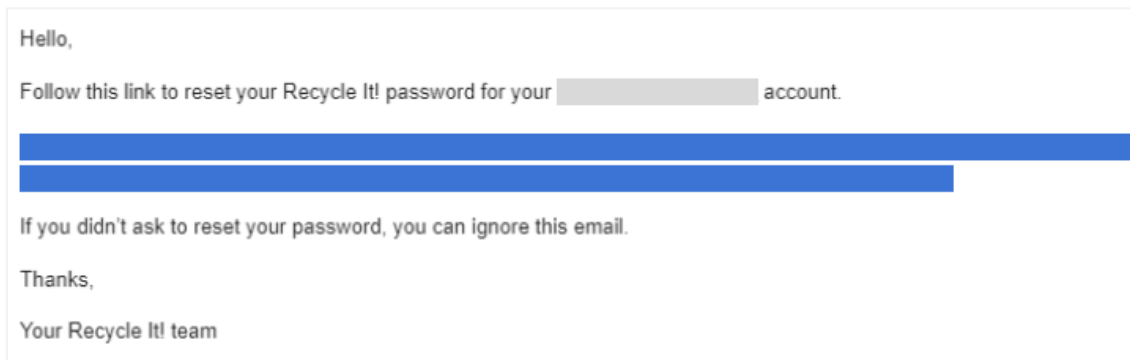


- Ανανέωση κωδικού στην περίπτωση που τον έχει ξεχάσει.



Εικόνα 30. Υπενθύμιση κωδικού σύνδεσης

Κατά την επαναφορά του κωδικού χρήστη, ο χρήστης καλείται να συμπληρώσει τη διεύθυνση ηλ. ταχυδρομείου του, στην οποία και του αποστέλλεται ένα μήνυμα που περιλαμβάνει οδηγίες σχετικές με την αλλαγή του προσωπικού του κωδικού.

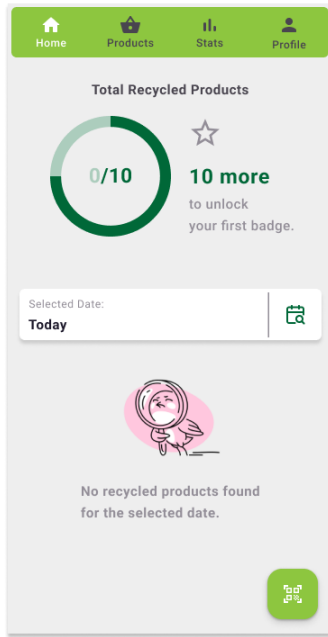


Εικόνα 31. Παράδειγμα email επαναφοράς κωδικού

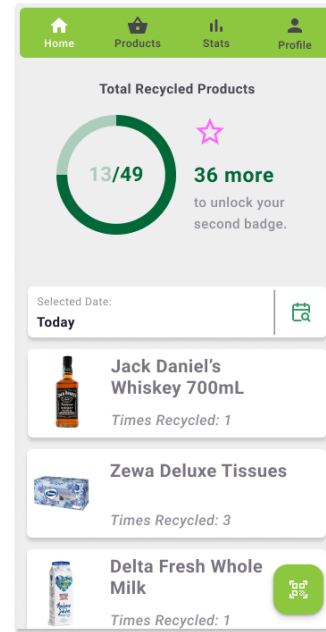
5.1.2. Αρχική Οθόνη

Μετά τη σύνδεσή του στην εφαρμογή, ο χρήστης, προτρέπεται στην *Αρχική οθόνη*, όπου ανάλογα με την προκαθορισμένη ημερομηνία (σήμερα) ή την επιλεγμένη από τον χρήστη, εμφανίζεται στην οθόνη το πλήθος των προϊόντων που έχουν ανακυκλωθεί από τον ίδιο, ή η επιγραφή «Δεν βρέθηκαν ανακυκλωμένα προϊόντα για την επιλεγμένη ημερομηνία»¹². Παρακάτω φαίνονται και οι δύο περιπτώσεις:



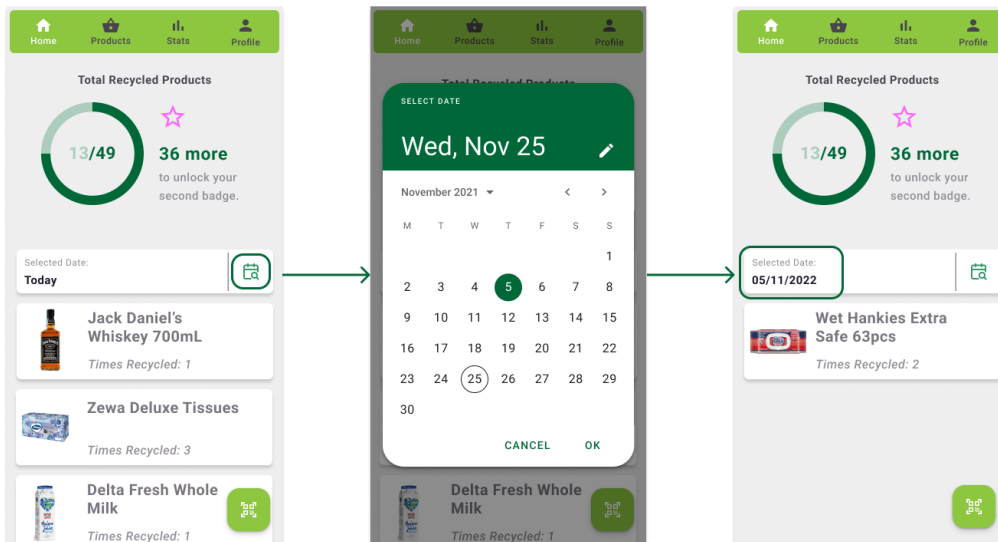


Εικόνα 32. Αρχική σελίδα χωρίς ανακυκλωμένα προϊόντα



Εικόνα 33. Αρχική σελίδα με προβολή ανακυκλωμένων προϊόντων

Για την επιλογή μίας ημερομηνίας διαφορετικής από την προκαθορισμένη, ο χρήστης καλείται να πατήσει στο εικονίδιο που βρίσκεται στα δεξιά του πλαισίου της επιλεγμένης ημερομηνίας, όπως φαίνεται και στις εικόνες παρακάτω. Πατώντας «OK» στο παράθυρο που προβάλλεται, θα παρατηρήσει τη λίστα των ανακυκλωμένων προϊόντων για την επιλεγμένη ημερομηνία.

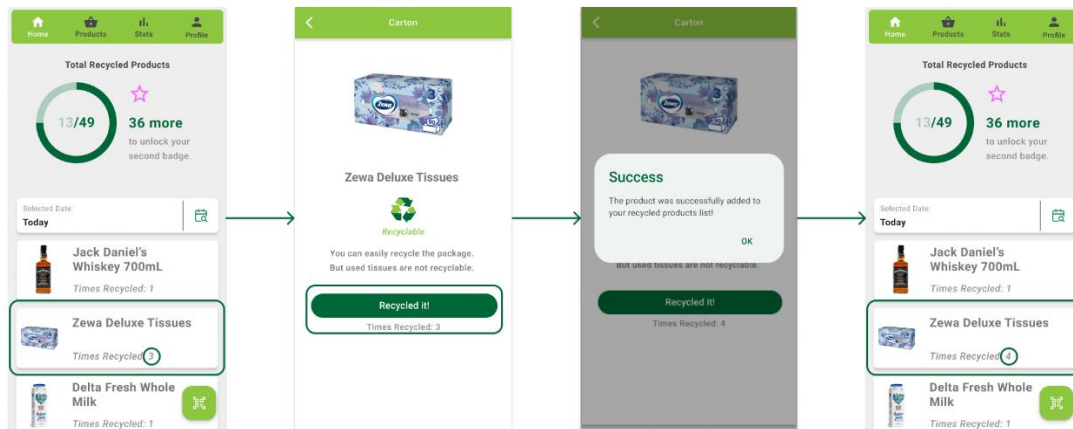


Εικόνα 34. Αναζήτηση προϊόντων ανά ημερομηνία

Εάν ο χρήστης επιλέξει κάποιο από τα προϊόντα που του εμφανίζονται, θα μεταβεί στην οθόνη *Λεπτομέρειες Προϊόντος*, που περιλαμβάνει στοιχεία για το συγκεκριμένο προϊόν, στα οποία περιλαμβάνονται και οδηγίες για τη σωστή ανακύκλωσή του. Επιπλέον, αν ο χρήστης

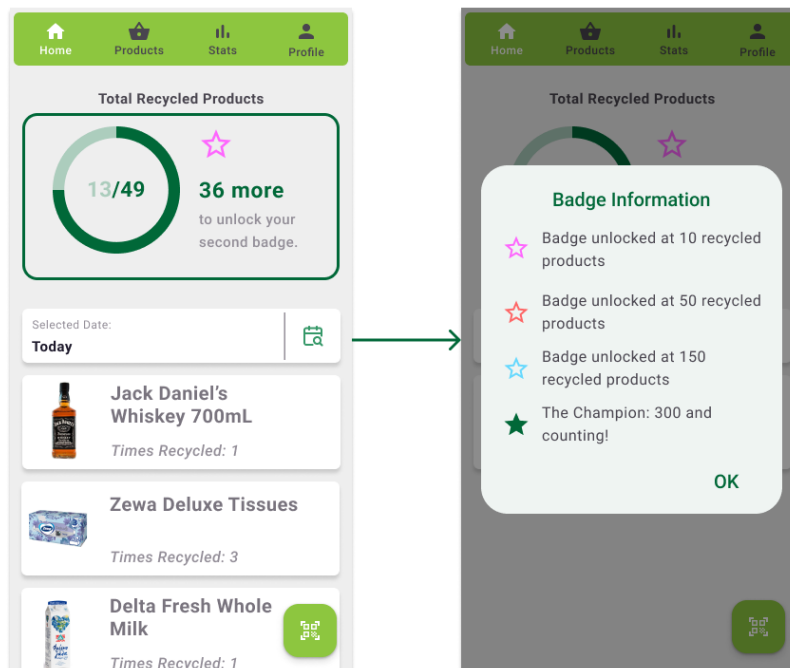


πατήσει το κουμπί με την ένδειξη «Recycled It!», η βάση δεδομένων ενημερώνεται και ανανεώνεται ο αριθμός ανακύκλωσης του προϊόντος στην παρούσα οθόνη, καθώς και στην Αρχική οθόνη, όταν επιστρέφει σε αυτήν.



Εικόνα 35. Καταχώρηση ανακυκλωμένου προϊόντος

Πέραν της προαναφερθείσας λειτουργικότητας, η εφαρμογή χρησιμοποιεί ένα σύστημα ποντοδότησης, κατά το οποίο ο χρήστης ανταμείβεται με πόντους μόλις ανακυκλώσει ένα προϊόν. Οι πόντοι αυτοί, αν συγκεντρωθούν, αντιστοιχούν σε μετάλλια, τα οποία ο χρήστης ξεκλειδώνει. Για να ενημερωθεί για τα διαθέσιμα μετάλλια, ο χρήστης αρκεί να πατήσει στον χώρο των στατιστικών που φαίνονται στο πάνω μέρος της αρχικής οθόνης.



Εικόνα 36. Πληροφορίες Μεταλλίων Χρήστη

Από εκεί ο χρήστης μπορεί να οδηγηθεί σε μία σειρά οθονών:

Η Αρχική οθόνη περιλαμβάνει ένα μενού τεσσάρων επιλογών στο άνω μέρος της οθόνης. Οι διαθέσιμες επιλογές παρατίθενται με τη σειρά (από αριστερά προς τα δεξιά) παρακάτω:

1. Αρχική οθόνη,



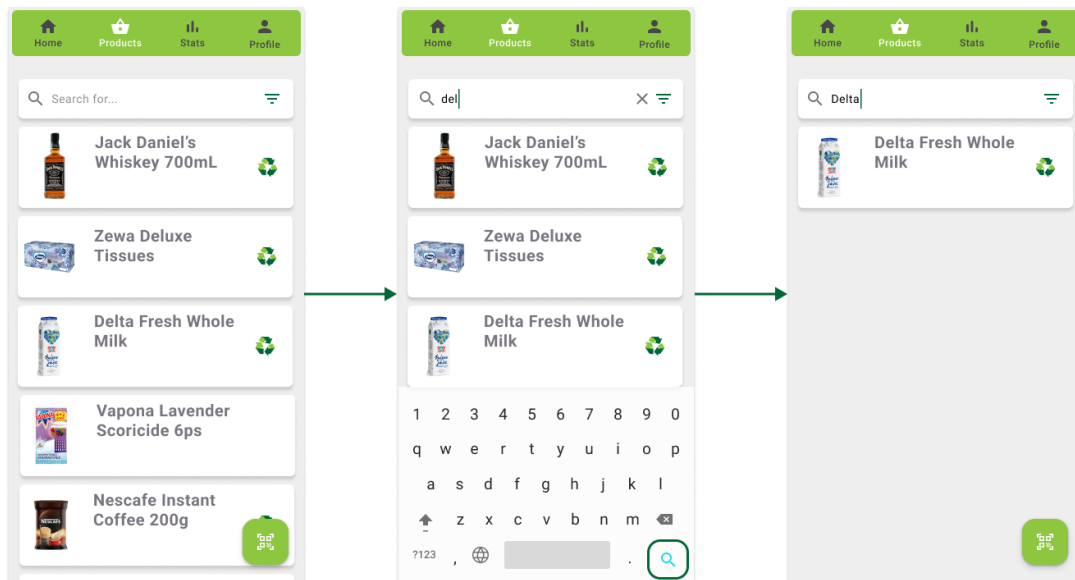
2. Προβολή συλλογής προϊόντων,
3. Προβολή γενικών στατιστικών ανακύκλωσης,
4. Προφίλ χρήστη.

Πέραν των επιλογών αυτών, υπάρχει και η δυνατότητα σάρωσης κωδικού barcode, η οποία βρίσκεται στην κάτω δεξιά πλευρά της οθόνης. Όλες θα περιγραφούν αναλυτικότερα στις επόμενες ενότητες.

5.1.3. Οθόνη προβολής συλλογής προϊόντων

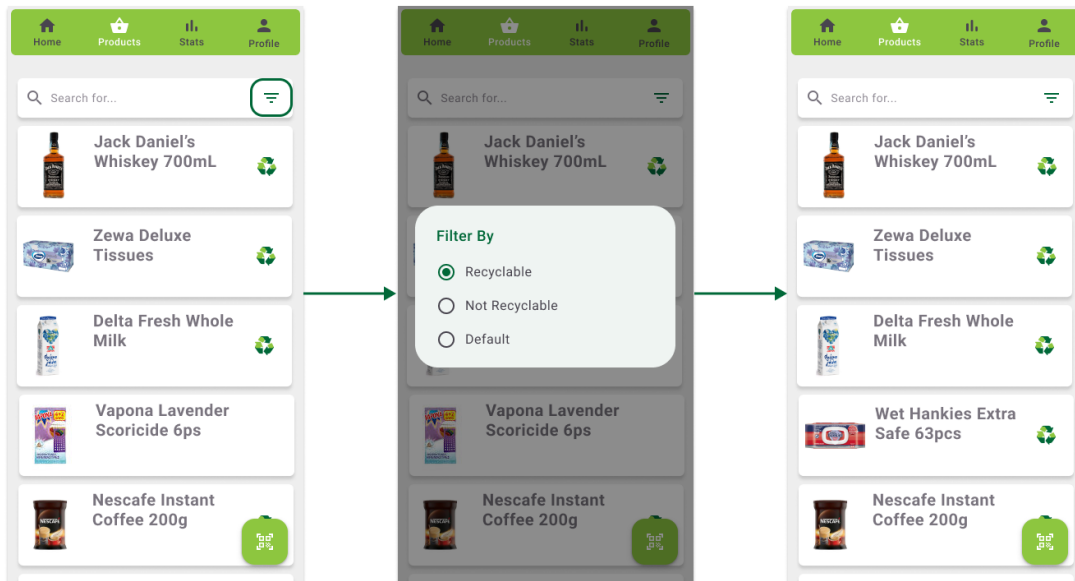
Στην οθόνη αυτή προβάλλεται το σύνολο των προϊόντων που υπάρχει στη βάση δεδομένων, μαζί με μία ένδειξη στα δεξιά του κάθε στοιχείου σχετικά με την ανακυκλωσιμότητά του ή μη. Παράλληλα, υποστηρίζονται οι παρακάτω επιλογές:

- Αναζήτηση προϊόντος με βάση το όνομά του.



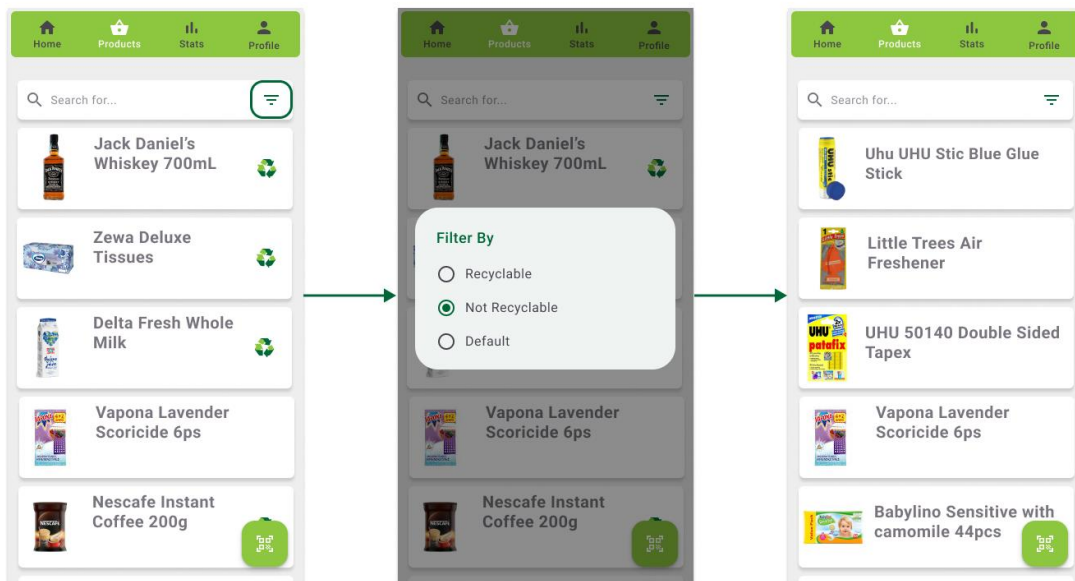
Εικόνα 37. Αναζήτηση προϊόντος

- Φιλτράρισμα της λίστας των προϊόντων ανάλογα με το αν είναι ανακυκλώσιμα ή μη (η προεπιλεγμένη διάταξη της λίστας περιλαμβάνει και τις δύο υποπεριπτώσεις):
 - Ανακυκλώσιμα



Εικόνα 38. Ενεργοποίηση φίλτρου προβολής ανακυκλώσιμων προϊόντων

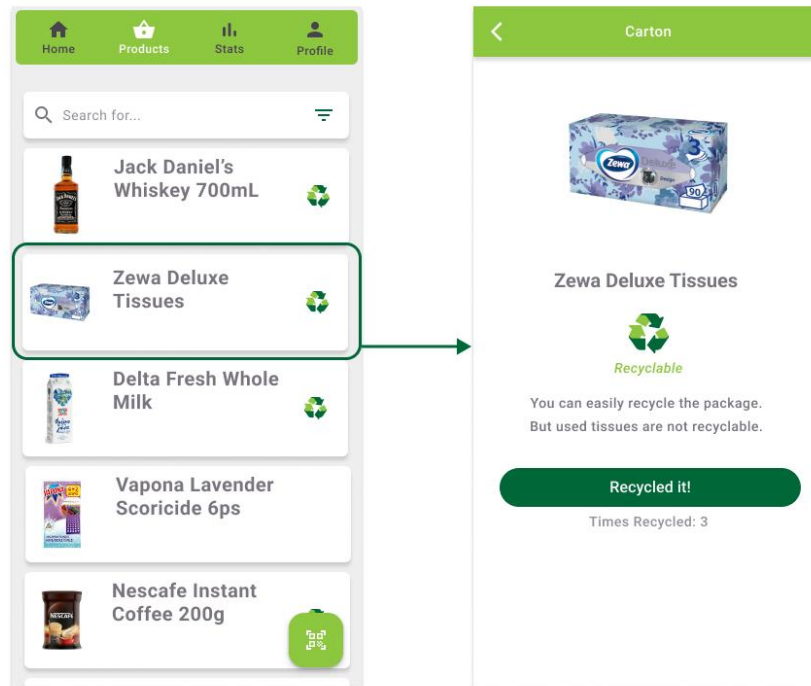
ο Μη Ανακυκλώσιμα



Εικόνα 39. Ενεργοποίηση φίλτρου προβολής μη ανακυκλώσιμων προϊόντων

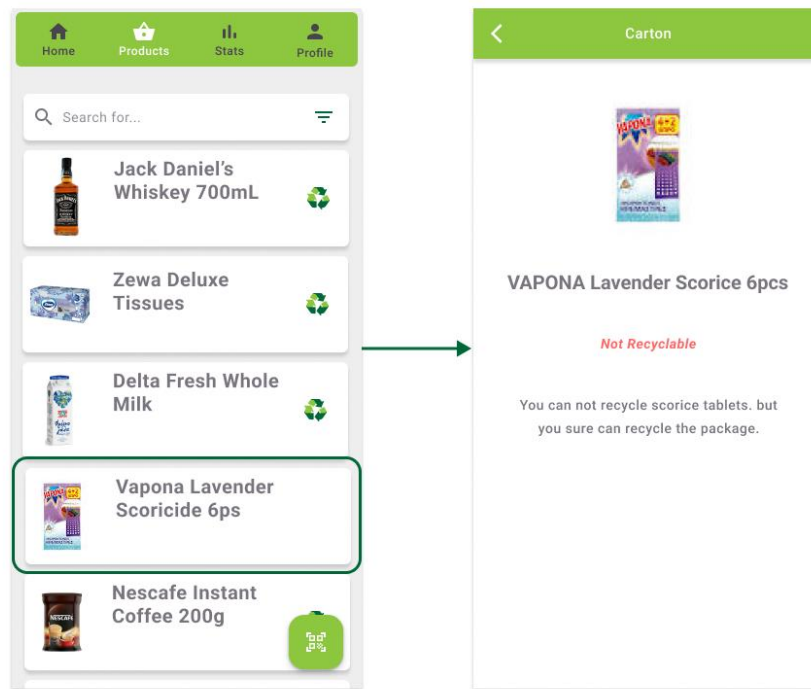


- Μετάβαση στην οθόνη Λεπτομέρειες Προϊόντος (Ανακυκλώσιμο προϊόν).



Εικόνα 40. Προβολή λεπτομερειών επιλεγμένου προϊόντος (Ανακυκλώσιμο)

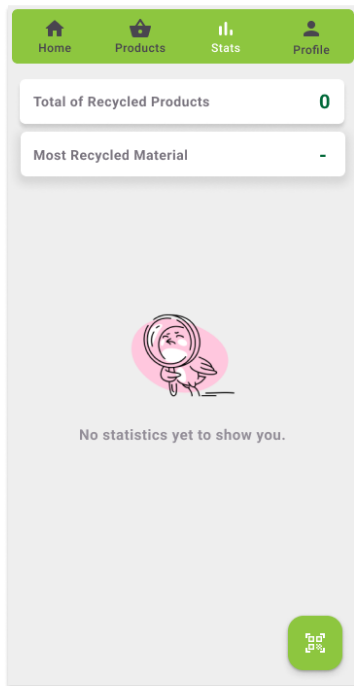
- Μετάβαση στην οθόνη Λεπτομέρειες Προϊόντος (Μη ανακυκλώσιμο προϊόν).



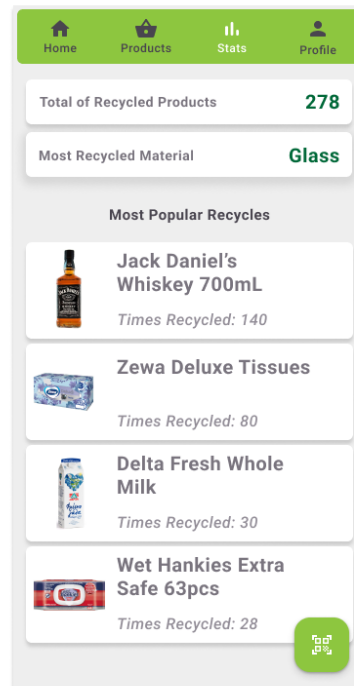
Εικόνα 41. Προβολή λεπτομερειών επιλεγμένου προϊόντος (Μη ανακυκλώσιμο)

5.1.4. Οθόνη προβολής στατιστικών

Ο χρήστης έχει τη δυνατότητα, αν το επιθυμεί, να μεταβεί στην οθόνη *Προβολής Στατιστικών*, όπου προβάλλεται το πιο συχνά ανακυκλωμένο υλικό από τους χρήστες της εφαρμογής, το σύνολο των προϊόντων που έχουν ανακυκλωθεί, καθώς και η λίστα των προϊόντων αυτών. Για την περίπτωση που δεν υπάρχουν δεδομένα, προβάλλεται το αντίστοιχο μήνυμα, όπως φαίνεται παρακάτω:



Εικόνα 42. Οθόνη στατιστικών (Χωρίς ύπαρξη δεδομένων)

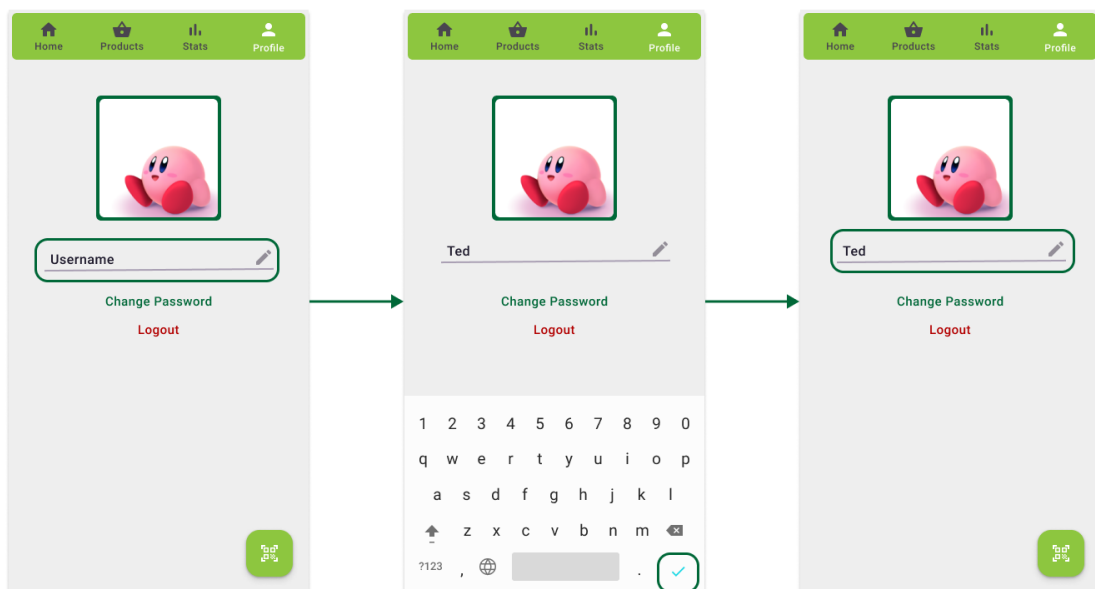


Εικόνα 43. Οθόνη στατιστικών (Με ύπαρξη δεδομένων)

5.1.5. Προφίλ Χρήστη

Στην οθόνη *Προφίλ* ο χρήστης έχει πλήθος δυνατοτήτων:

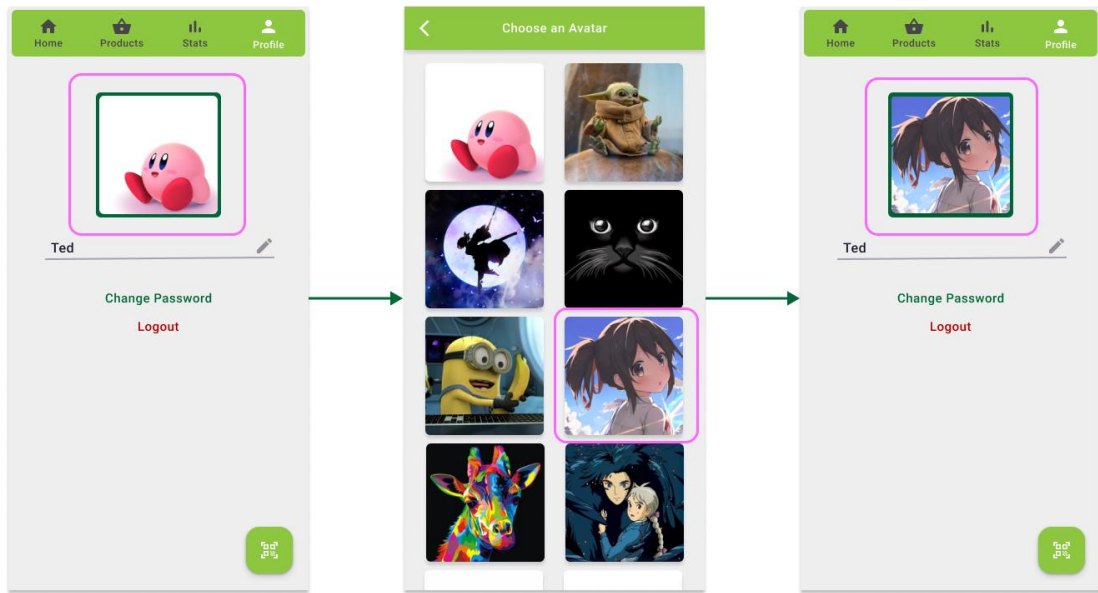
- Συμπλήρωση/ Ενημέρωση του username του.



Εικόνα 44. Καταχώρηση username

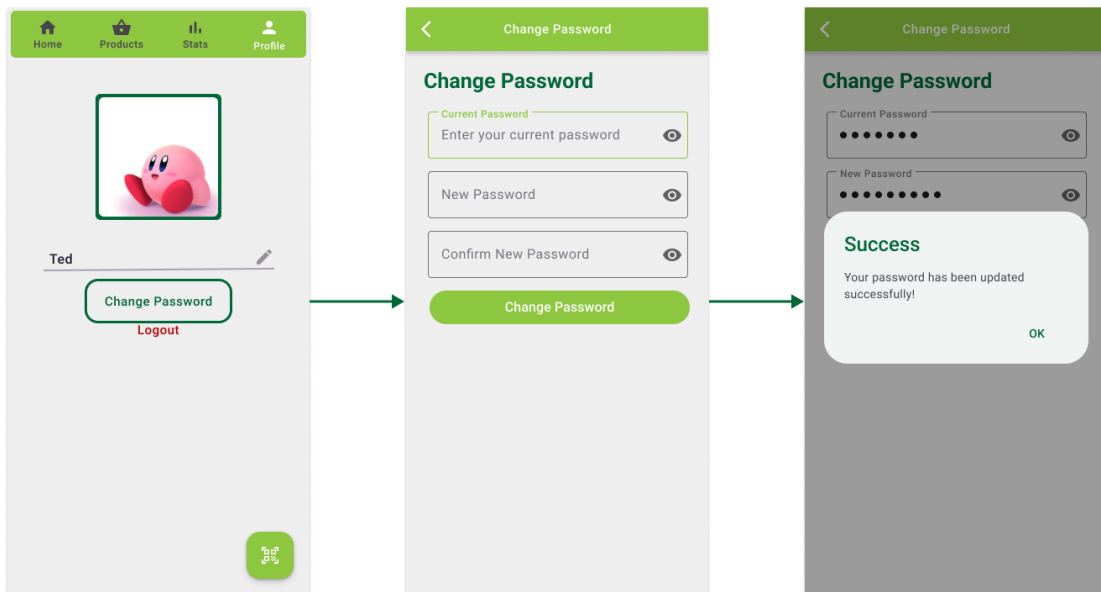


- Αλλαγή του avatar του.



Εικόνα 45. Αλλαγή εικόνας προφίλ

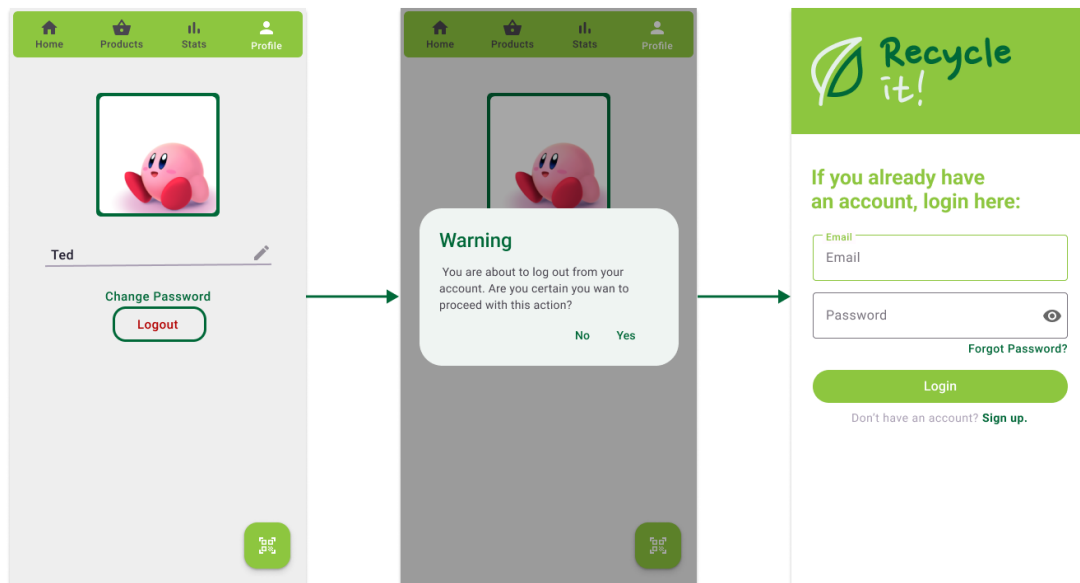
- Αλλαγή του προσωπικού του κωδικού.



Εικόνα 46. Αλλαγή κωδικού σύνδεσης

- Αποσύνδεση από την εφαρμογή.



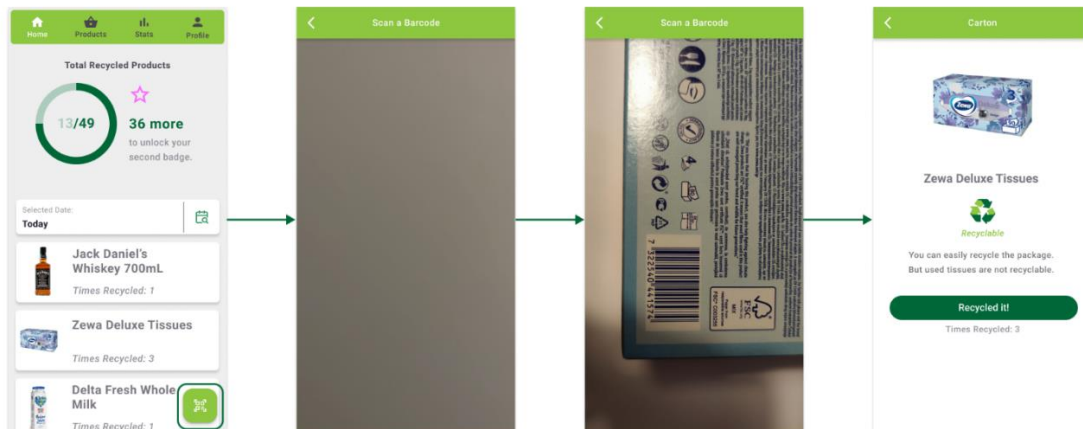


Εικόνα 47. Αποσύνδεση από την εφαρμογή

5.1.6. Οθόνη αναγνώρισης κωδικού barcode

Μετά την αναγνώριση ενός κωδικού barcode, ο κωδικός αυτός συγκρίνεται με τους κωδικούς που υπάρχουν στη βάση δεδομένων. Αν υπάρξει αντιστοιχία, ο χρήστης μεταβαίνει στην οθόνη *Λεπτομέρειες Προϊόντος*, διαφορετικά, εμφανίζεται αντίστοιχο μήνυμα. Οι δύο περιπτώσεις φαίνονται παρακάτω:

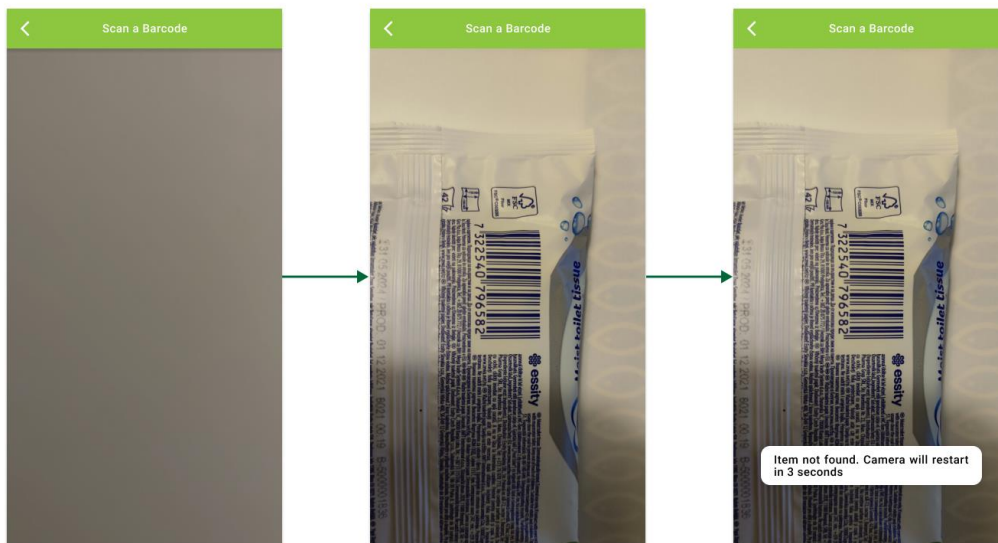
- Αναγνώριση προϊόντος.



Εικόνα 48. Επιτυχής αναγνώριση προϊόντος με barcode scanning

- Αποτυχία αναγνώρισης προϊόντος.





Εικόνα 49. Αποτυχία ταύτισης προϊόντος με τον αναγνωρισθέντα κωδικό barcode

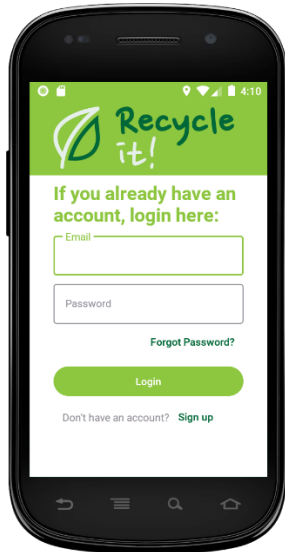
5.2. Υλοποίηση Εφαρμογής

Σε αυτή την ενότητα παρουσιάζεται η εφαρμογή μέσω στιγμιότυπων οθόνης που συλλέχθηκαν από συσκευές διαφορετικών διαστάσεων. Ακολουθεί η λίστα τους:

- Nexus S. Διαστάσεις: 4,65" - 480x800,
- Galaxy Nexus. Διαστάσεις: 4,65" - 720x1280,
- Pixel 2XL. Διαστάσεις: 5,99" - 1440x2880,
- Pixel 4XL. Διαστάσεις: 6,3" - 1440x3040

Ο έλεγχος της εφαρμογής στις παραπάνω συσκευές πραγματοποιήθηκε προκειμένου να διαπιστωθεί αν οι οθόνες προβάλλονται εξίσου καλά σε κάθε μία από αυτές, παρά τη διαφορά στις διαστάσεις τους. Τα αποτελέσματα είναι διαθέσιμα παρακάτω.

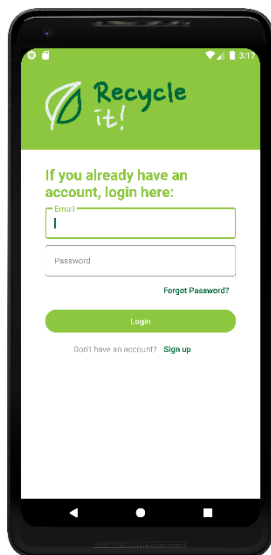
5.2.1. Οθόνη Σύνδεσης



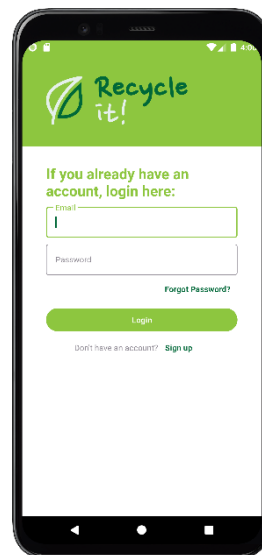
Εικόνα 50. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Nexus S



Εικόνα 51. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Galaxy Nexus



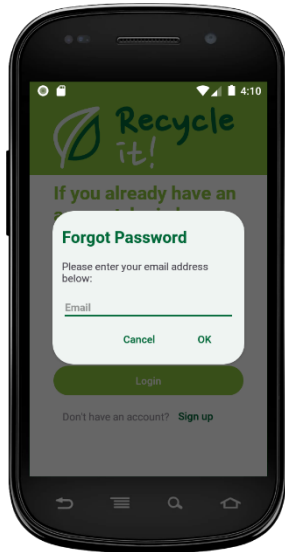
Εικόνα 52. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Pixel 2XL



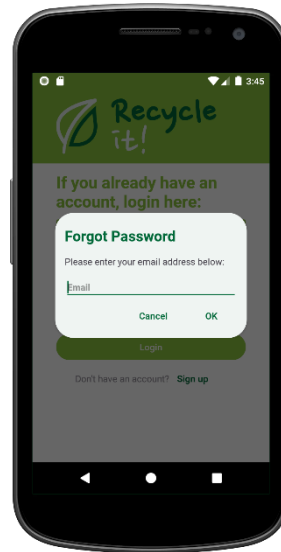
Εικόνα 53. Στιγμιότυπο οθόνης Σύνδεσης - Συσκευή Pixel 4XL



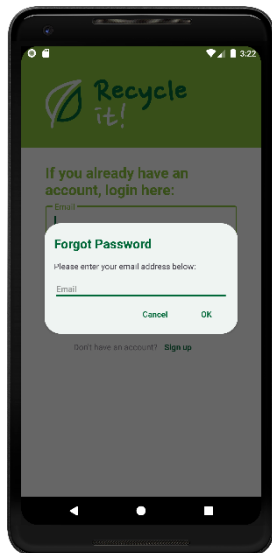
5.2.1.1. Παράθυρο Υπενθύμισης Κωδικού



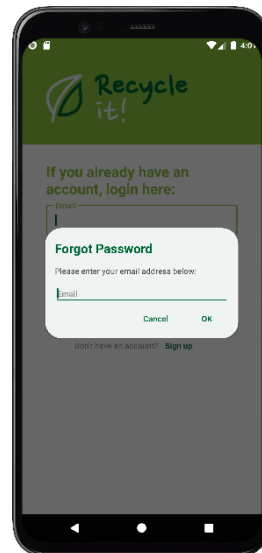
Εικόνα 54. Στιγμιότυπο Παράθρου Υπενθύμισης Κωδικού - Συσκευή Nexus S



Εικόνα 55. Στιγμιότυπο Παράθρου Υπενθύμισης Κωδικού - Συσκευή Galaxy Nexus

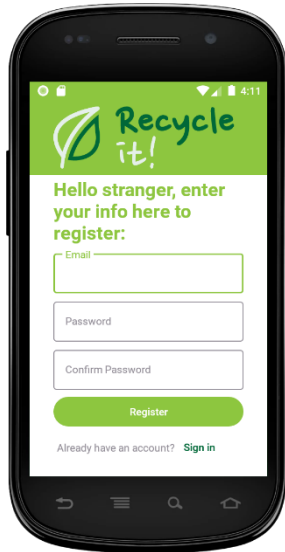


Εικόνα 56. Στιγμιότυπο Παράθρου Υπενθύμισης Κωδικού - Συσκευή Pixel 2XL

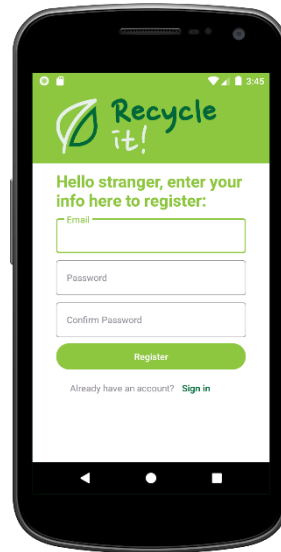


Εικόνα 57. Στιγμιότυπο Παράθρου Υπενθύμισης Κωδικού - Συσκευή Pixel 4XL

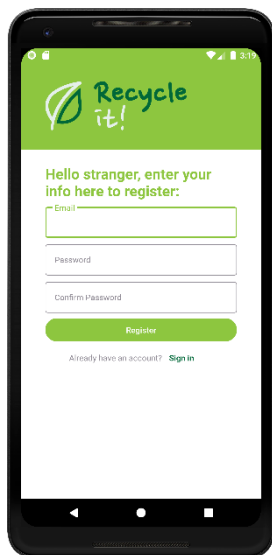
5.2.2. Οθόνη Εγγραφής



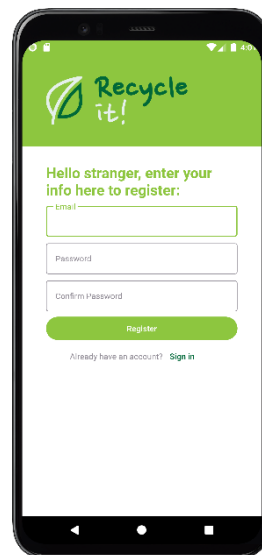
Εικόνα 58. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Nexus S



Εικόνα 59. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Galaxy Nexus



Εικόνα 60. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Pixel 2XL

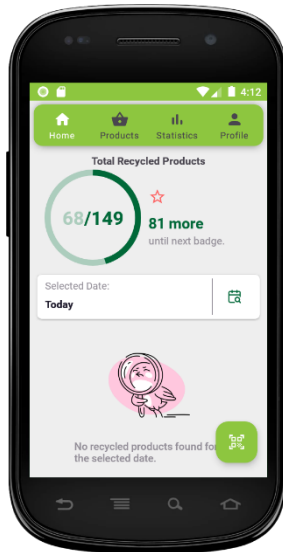


Εικόνα 61. Στιγμιότυπο οθόνης Εγγραφής - Συσκευή Pixel 4XL

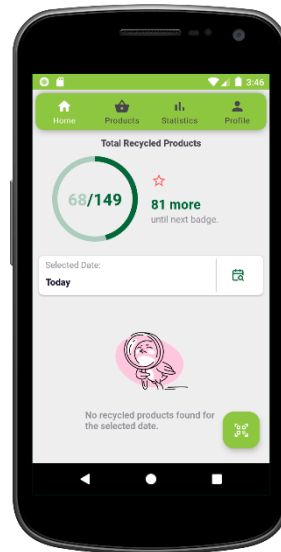


5.2.3. Αρχική Οθόνη

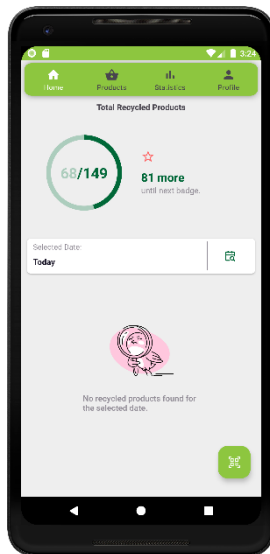
5.2.3.1. Χωρίς Ανακυκλωμένα Προϊόντα



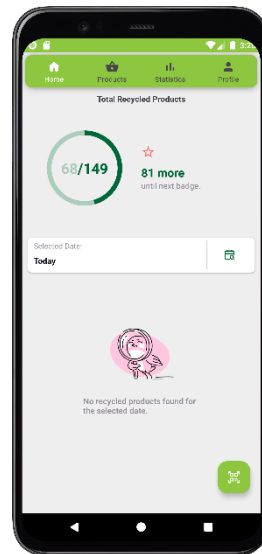
Εικόνα 62. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Nexus S



Εικόνα 63. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Galaxy Nexus



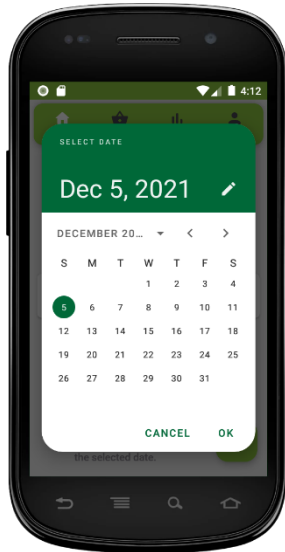
Εικόνα 64. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Pxeil 2XL



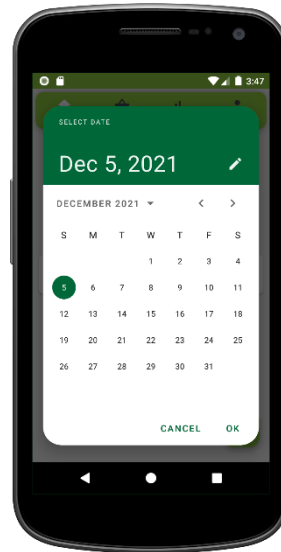
Εικόνα 65. Στιγμιότυπο Αρχικής Οθόνης Χωρίς Ανακυκλωμένα Προϊόντα - Συσκευή Pixel 4XL



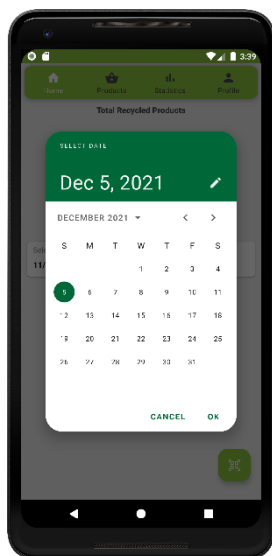
5.2.3.2. Παράθυρο Επιλογής Ημερομηνίας



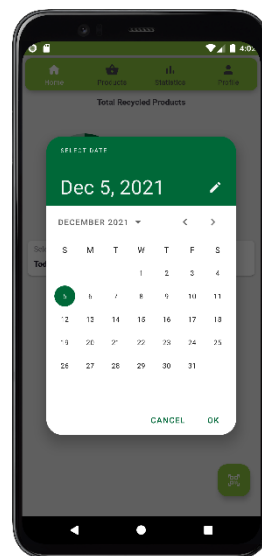
Εικόνα 66. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Nexus S



Εικόνα 67. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Galaxy Nexus



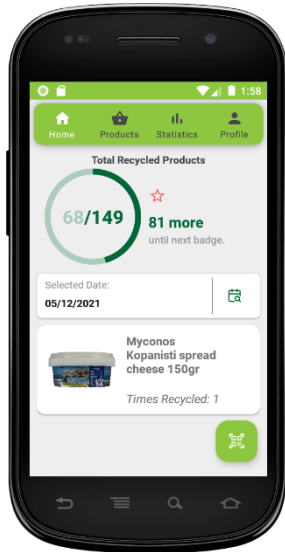
Εικόνα 68. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Pixel 2XL



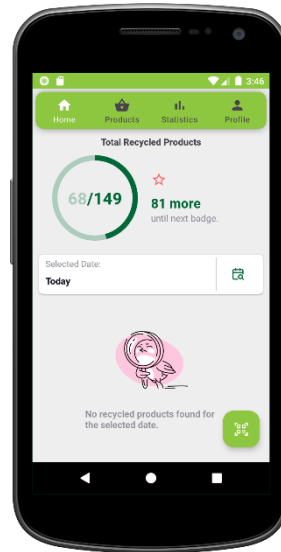
Εικόνα 69. Στιγμιότυπο Παραθύρου Επιλογής Ημερομηνίας - Συσκευή Pixel 4XL



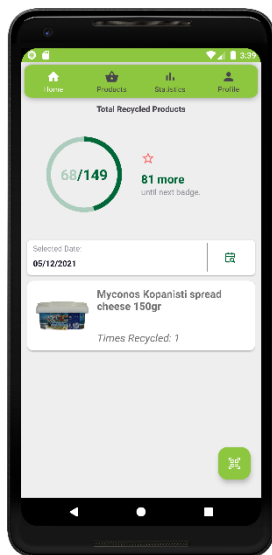
5.2.3.3. Με Ανακυκλωμένα Προϊόντα



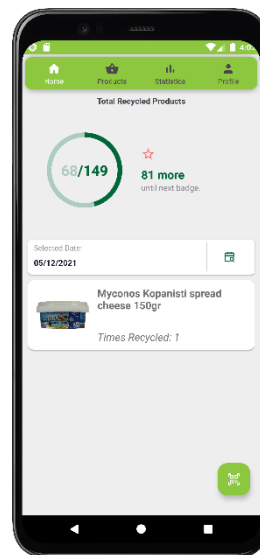
Εικόνα 70. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Nexus S



Εικόνα 71. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Galaxy Nexus



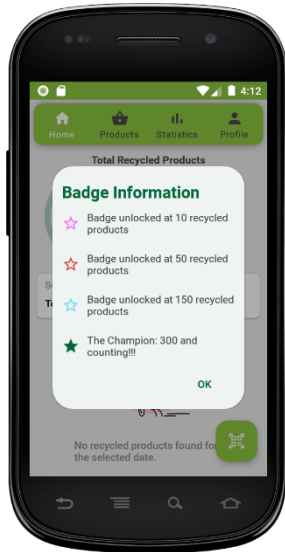
Εικόνα 72. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Pixel 2XL



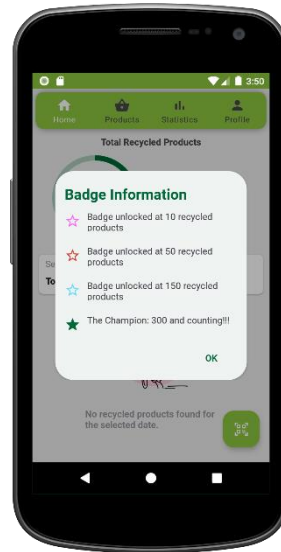
Εικόνα 73. Στιγμιότυπο Αρχικής Οθόνης με Ανακυκλωμένα Προϊόντα - Συσκευή Pixel 4XL



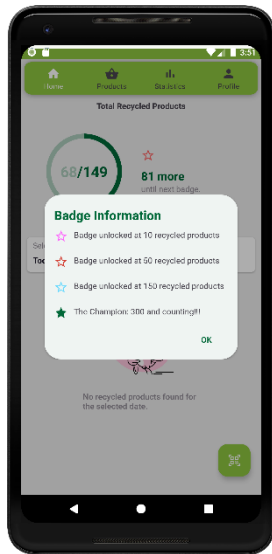
5.2.3.4. Παράθυρο Πληροφοριών για τα Μετάλλια



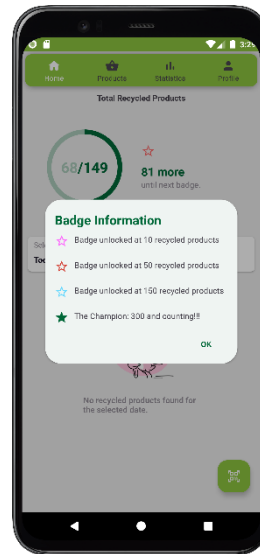
Εικόνα 74. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Nexus S



Εικόνα 75. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Galaxy Nexus



Εικόνα 76. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Pixel 2XL

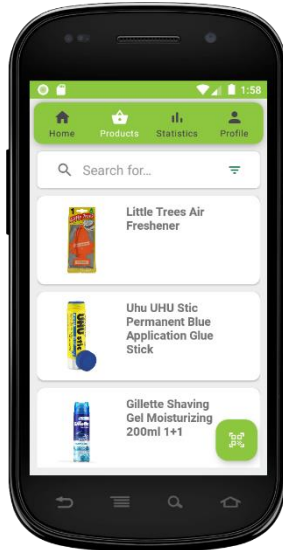


Εικόνα 77. Στιγμιότυπο Παραθύρου Πληροφοριών για τα Μετάλλια - Συσκευή Pixel 4XL

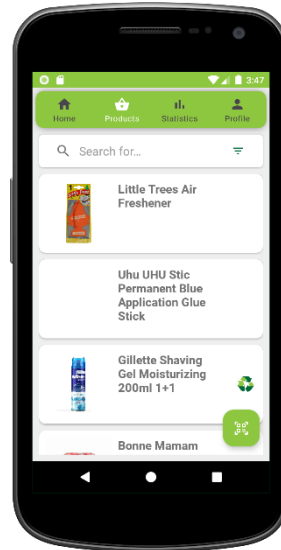


5.2.4. Οθόνη Προβολής Προϊόντων

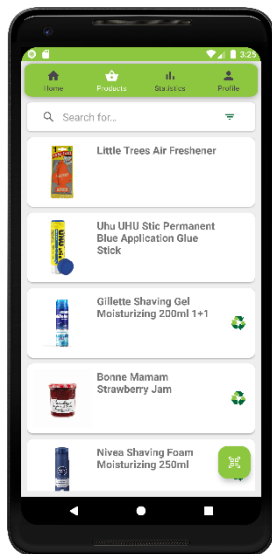
5.2.4.1. Προεπιλογή



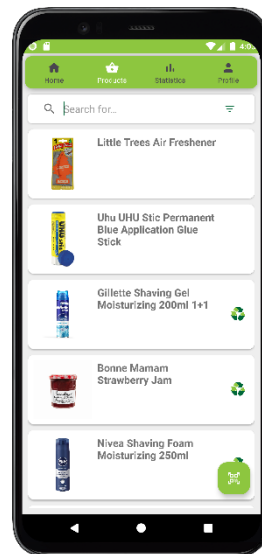
Εικόνα 78. Στιγμιότυπο Οθόνης Προβολή Προϊόντων - Συσκευή Nexus S



Εικόνα 79. Στιγμιότυπο Οθόνης Προβολή Προϊόντων - Συσκευή Galaxy Nexus



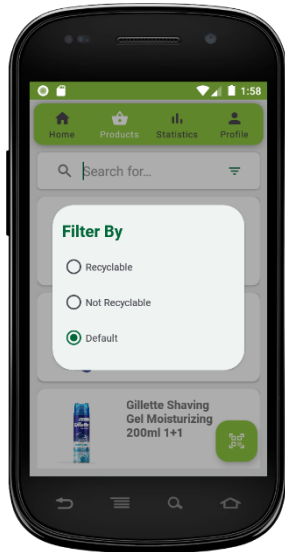
Εικόνα 80. Στιγμιότυπο Οθόνης Προβολή Προϊόντων - Συσκευή Pixel 2XL



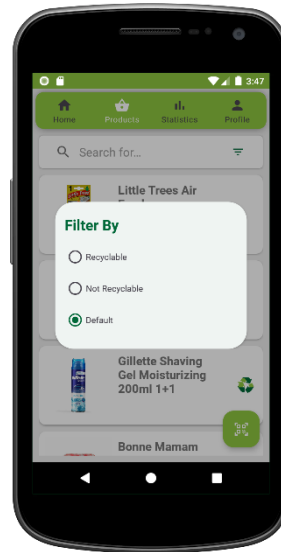
Εικόνα 81. Στιγμιότυπο Οθόνης Προβολή Προϊόντων - Συσκευή Pixel 4XL



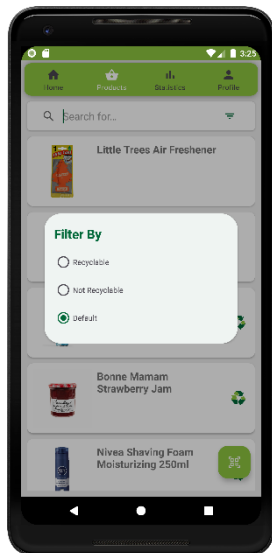
5.2.4.2. Παράθυρο Επιλογής Φίλτρου Προβολής



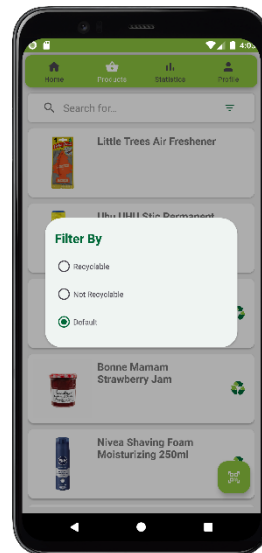
Εικόνα 82. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής - Συσκευή Nexus S



Εικόνα 83. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής - Συσκευή Galaxy Nexus



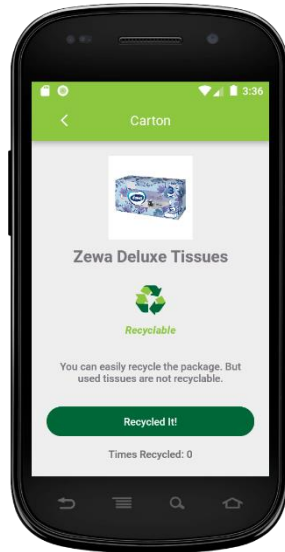
Εικόνα 84. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής Προϊόντων - Συσκευή Pixel 2XL



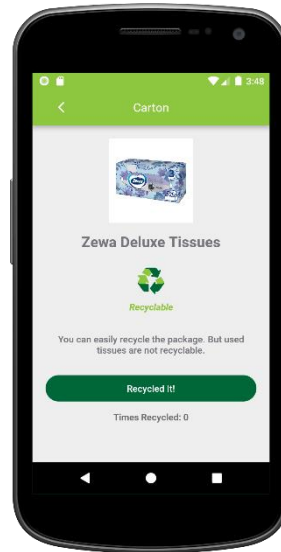
Εικόνα 85. Στιγμιότυπο Παραθύρου Φίλτρου Προβολής Προϊόντων - Συσκευή Pixel 4XL



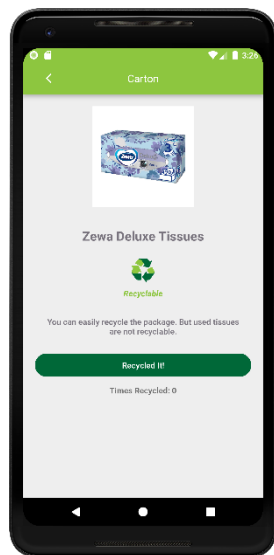
5.2.5. Οθόνη Προβολής Λεπτομερειών Προϊόντος



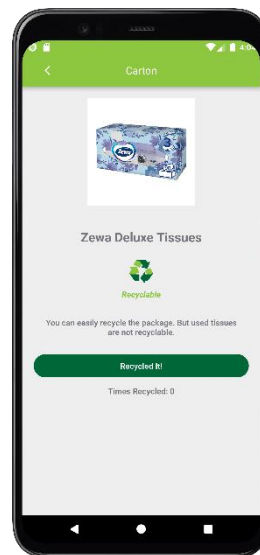
Εικόνα 86. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Nexus S



Εικόνα 87. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Galaxy Nexus



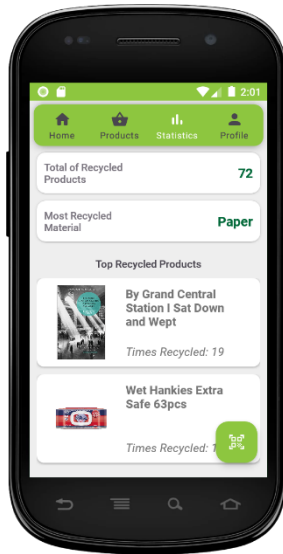
Εικόνα 88. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Pixel 2XL



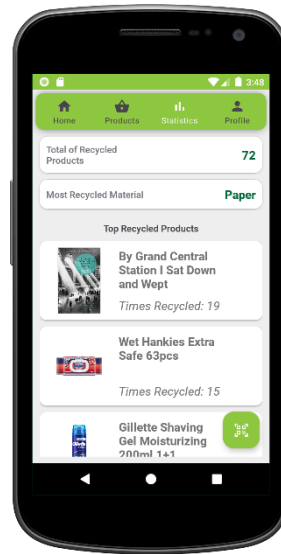
Εικόνα 89. Στιγμιότυπο Οθόνης Προβολής Λεπτομερειών Προϊόντος - Συσκευή Pixel 4XL



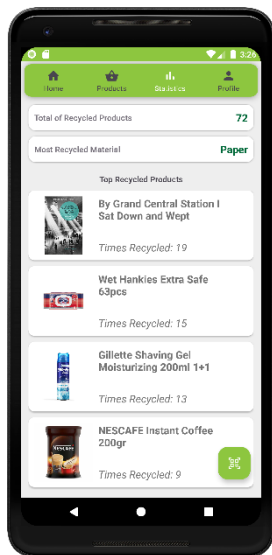
5.2.6. Οθόνη Προβολής Στατιστικών



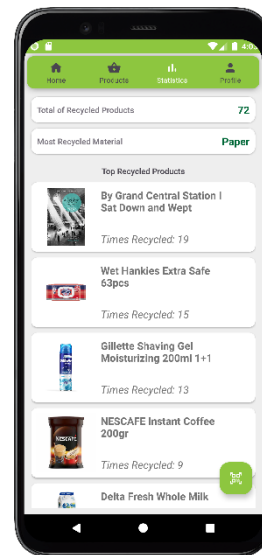
Εικόνα 90. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Nexus S



Εικόνα 91. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Galaxy Nexus

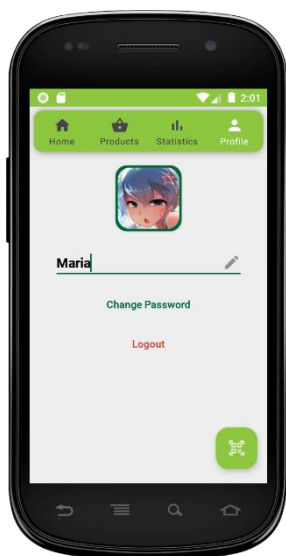


Εικόνα 92. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Pixel 2XL

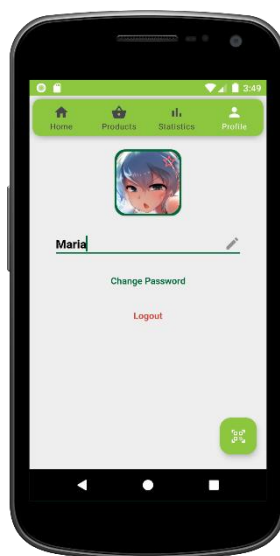


Εικόνα 93. Στιγμιότυπο Οθόνης Προβολής Στατιστικών - Συσκευή Pixel 4XL

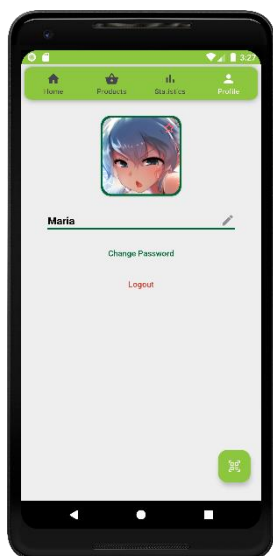
5.2.7. Οθόνη Προφίλ



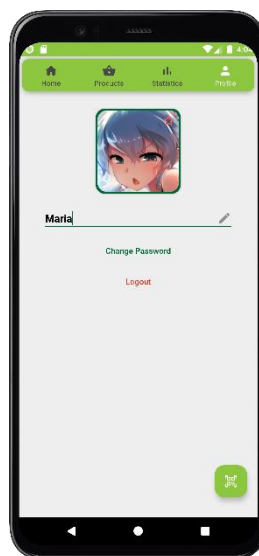
Εικόνα 94. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Nexus S



Εικόνα 95. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Galaxy Nexus



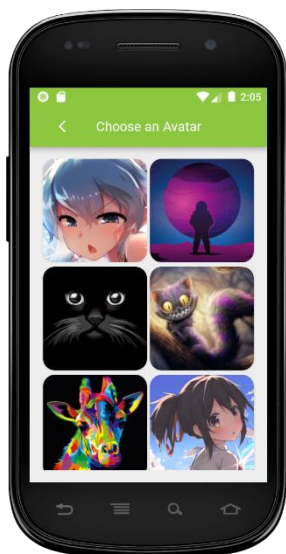
Εικόνα 96. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Pixel 2XL



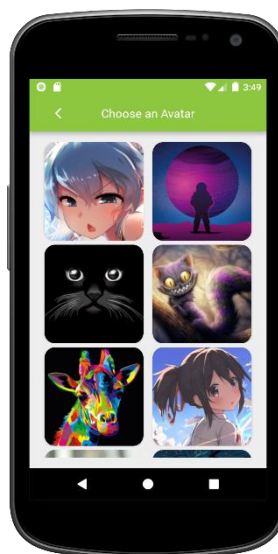
Εικόνα 97. Στιγμιότυπο Οθόνης Προφίλ - Συσκευή Pixel 4XL



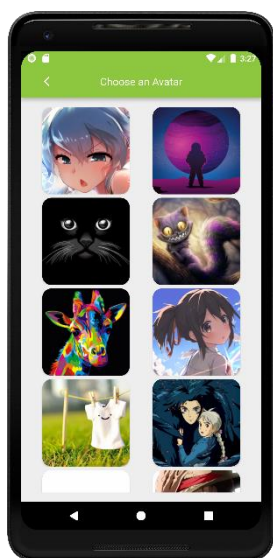
5.2.7.1. Αλλαγή Άβαταρ



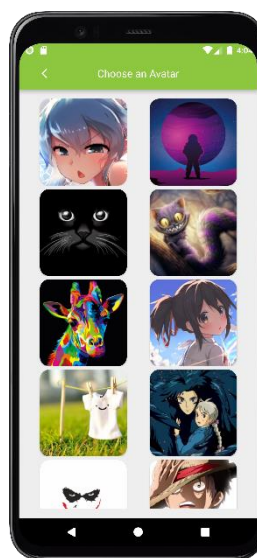
Εικόνα 98. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Nexus S



Εικόνα 99. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Galaxy Nexus



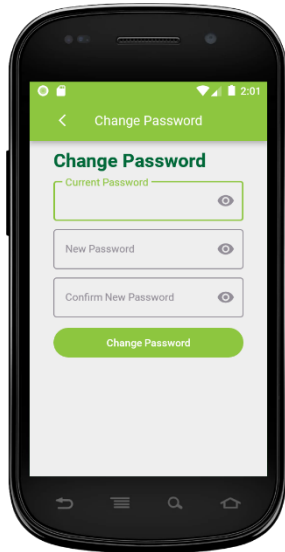
Εικόνα 100. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Pixel 2XL



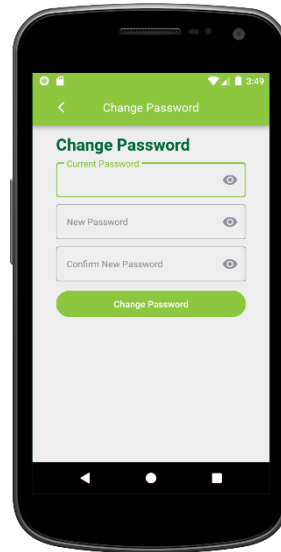
Εικόνα 101. Στιγμιότυπο Οθόνης Αλλαγής Άβαταρ - Συσκευή Pixel 4XL



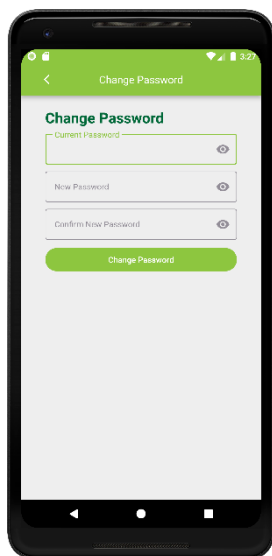
5.2.7.2. Αλλαγή Κωδικού



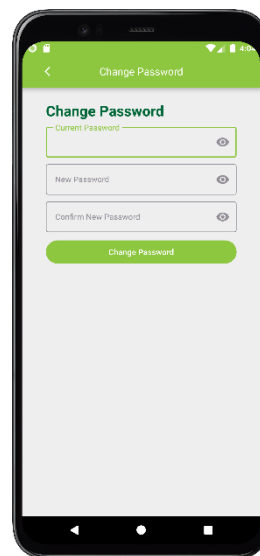
Εικόνα 102. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Nexus S



Εικόνα 103. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Galaxy Nexus



Εικόνα 104. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Pixel 2XL



Εικόνα 105. Στιγμιότυπο Οθόνης Αλλαγής Κωδικού - Συσκευή Pixel 4XL



5.2.8. Οθόνη Σάρωσης Κωδικού Προϊόντος



Εικόνα 106. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος - Συσκευή Nexus S



Εικόνα 107. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος - Συσκευή Galaxy Nexus



Εικόνα 108. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος - Συσκευή Pixel 2XL



Εικόνα 109. Στιγμιότυπο Οθόνης Σάρωσης Κωδικού Προϊόντος

6. Σχεδιασμός

6.2. Γλώσσες Προγραμματισμού

6.2.1. Kotlin^{13 14}

Πρόκειται για μία αντικειμενοστραφή γλώσσα προγραμματισμού, η οποία αναπτύχθηκε από την JetBrains, γνωστή για το IntelliJ IDEA. Η επίσημη κυκλοφορία της ήταν τον Φεβρουάριο του 2016 και έχει οριστεί από την Google ως η επίσημη γλώσσα ανάπτυξης Android εφαρμογών.



Η Kotlin βασίζεται στην γλώσσα προγραμματισμού Java και αναπτύχθηκε με σκοπό να προσφέρει μία βελτιωμένη εμπειρία στον χρήστη, λαμβάνοντας υπόψιν τα μειονεκτήματα της δεύτερης. Υποστηρίζει τη διαλειτουργικότητα μεταξύ των δύο γλωσσών παρέχοντας σαφές πλεονέκτημα σε πιθανή αλλαγή της γλώσσας ενός προγράμματος (από Java σε Kotlin), ενώ παράλληλα διατείνεται ότι είναι ασφαλέστερη και απλούστερη. Μερικά ακόμα στοιχεία τα οποία προσάπτονται στη φιλοσοφία της εν λόγω γλώσσας είναι η ευκολότερη ανάγνωσή της, η φιλικότητα προς τον χρήστη, η επαναχρησιμοποίηση του υπάρχοντος κώδικα και τέλος η αποδοτικότητα (Marcin Moskala, 2020).

6.3. Εργαλεία

Για την υλοποίηση της παρούσας εφαρμογής χρησιμοποιήθηκε πλήθος εργαλείων τα οποία παρατίθενται αναλυτικά παρακάτω.

6.3.1. Android Studio¹⁵

Το Android Studio είναι το επίσημο ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) για την δημιουργία Android εφαρμογών, βασισμένο στο πρόγραμμα IntelliJ IDEA. Πέρα από τα βασικά, όπως ο code editor και απαραίτητα εργαλεία, περιλαμβάνει μία σειρά από διευκολύνσεις. Για παράδειγμα, έναν εξομοιωτή Android συσκευής για τη δοκιμή της εφαρμογής υπό ανάπτυξη (*Emulator*), συνεργασία με το GitHub ώστε να είναι πιο εύκολη και γρήγορη η μεταφορά και αποθήκευση κώδικα, καθώς και ο διαμοιρασμός του μεταξύ των εμπλεκόμενων προσώπων. Παράλληλα, υπάρχει υποστήριξη για τεστ του κώδικα και της λειτουργίας της εφαρμογής, ενώ επίσης είναι δυνατός ο έλεγχος της απόδοσης της εφαρμογής (*Profiler*).

6.3.2. Firebase¹⁶

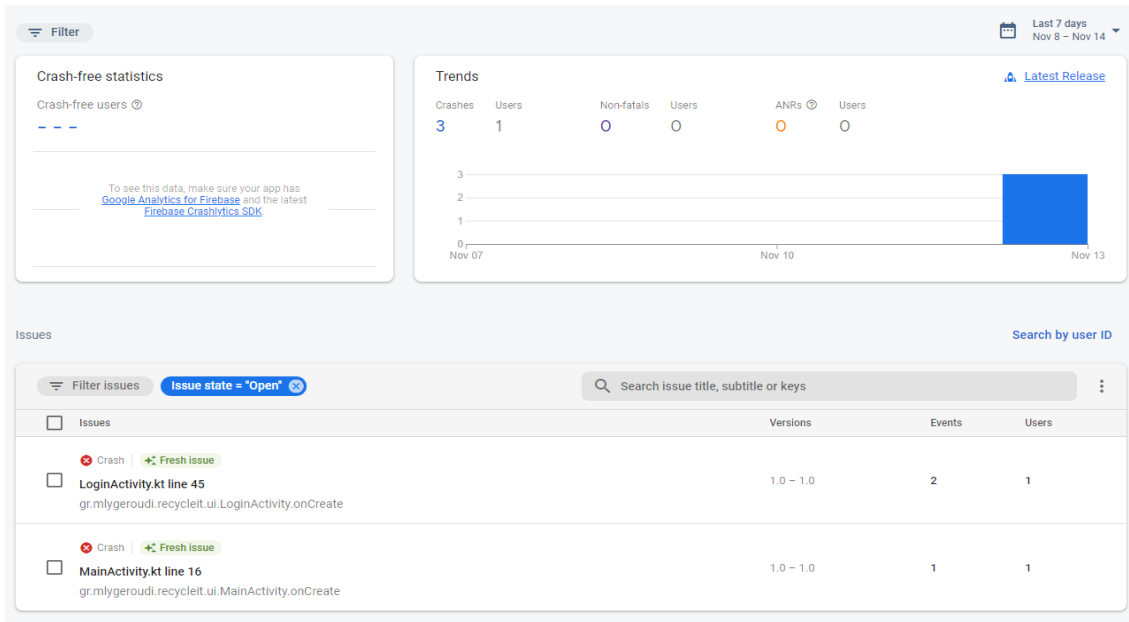
Η Firebase αποτελεί μία εναλλακτική στον θεσμοθετημένο τρόπο λειτουργίας του μέσου προγραμματιστή. Προσφέρει τη δυνατότητα της ενασχόλησης με πιο ουσιαστικά ζητήματα, που αφορούν αποκλειστικά τη λειτουργικότητα της εφαρμογής υπό ανάπτυξη, παρέχοντας εργαλεία όπως αναλυτικά στατιστικά εξαιρετικά χρήσιμα για την καλή λειτουργία της εφαρμογής και τη βελτίωσή της (π.χ. το Crashlytics, το οποίο βοηθά στην απομακρυσμένη επίβλεψη της, καταγράφοντας logs για το κάθε crash που πραγματοποιείται), βάσεις δεδομένων, εγγραφή και επιβεβαίωση στοιχείων κατά τη σύνδεση ενός χρήστη (*Authentication*), ανταλλαγή μηνυμάτων και πολλά ακόμα. Οι υπηρεσίες αυτές γίνονται μέσω Cloud και η χρήση τους είναι αρκετά απλή και εύκολη. Παρακάτω αναφέρονται όσα χρησιμοποιήθηκαν στην εφαρμογή που υλοποιήσαμε.

6.2.2.1. Firebase Crashlytics¹⁷

Η υπηρεσία Crashlytics βοηθά στην απομακρυσμένη επίβλεψη των σφαλμάτων που ενδέχεται να προκύψουν κατά τη λειτουργία της εφαρμογής. Εάν η εφαρμογή σταματήσει να λειτουργεί λόγω κάποιου σφάλματος, τότε η εν λόγω υπηρεσία διατηρεί μία λίστα από στοιχεία χρήσιμα για τη διερεύνηση του σφάλματος, όπως την ημερομηνία και ώρα που αυτό πραγματοποιήθηκε, την έκδοση της εφαρμογής, στοιχεία για το λειτουργικό, λόγου χάρη η έκδοση του Android, καθώς επίσης και στοιχεία για τη συσκευή όπου παρατηρήθηκε το περιστατικό, π.χ. *Brand: Xiaomi, Model: Mi 10T Pro*. Με το επόμενο άνοιγμα της εφαρμογής, τα στοιχεία αυτά αποστέλλονται και είναι εμφανή μέσω της πλατφόρμας της Firebase.

Η υπηρεσία αυτή είναι πραγματικά χρήσιμη για οποιονδήποτε προγραμματιστή, διότι μπορεί να τροποποιεί την εφαρμογή του με απομακρυσμένη ανατροφοδότηση, ανά πάσα στιγμή. Ενώ ακόμα, έχει πρόσβαση στη συχνότητα του κάθε σφάλματος, έχοντας έτσι τη δυνατότητα προτεραιοποίησης της αντιμετώπισης των σφαλμάτων. Παράδειγμα της χρήσης της παρατίθεται παρακάτω:





Εικόνα 110. Αρχική οθόνη Crashlytics



Event summary 1.0 (1) 12 Mi 10T Pro Nov 14, 2022, 1:42:56 AM

Stack trace Keys Logs Data

TXT

Fatal Exception: java.lang.RuntimeException
Unable to start activity ComponentInfo{gr.mlygeroudi.recycleit/gr.mlygeroudi.recycleit.ui.LoginActivity}: java.lang.Exception: This is a test!

```

android.app.ActivityThread.performLaunchActivity (ActivityThread.java:3763)
android.app.ActivityThread.handleLaunchActivity (ActivityThread.java:3937)
android.app.servertransaction.LaunchActivityItem.execute (LaunchActivityItem.java:103)
android.app.servertransaction.TransactionExecutor.executeCallbacks (TransactionExecutor.java:135)
android.app.servertransaction.TransactionExecutor.execute (TransactionExecutor.java:95)
android.app.ActivityThread$H.handleMessage (ActivityThread.java:2288)
android.os.Handler.dispatchMessage (Handler.java:106)
android.os.Looper.loopOnce (Looper.java:210)
android.os.Looper.loop (Looper.java:299)
android.app.ActivityThread.main (ActivityThread.java:8293)
java.lang.reflect.Method.invoke (Method.java)
com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run (RuntimeInit.java:556)
com.android.internal.os.ZygoteInit.main (ZygoteInit.java:1045)

```

Caused by java.lang.Exception
This is a test! gr.mlygeroudi.recycleit.ui.LoginActivity.onCreate

Εικόνα 111. Επισκόπηση σφάλματος

Event summary 1.0 (1) 12 Mi 10T Pro Nov 14, 2022, 1:42:56 AM

Stack trace Keys Logs Data

| Device | Operating System | Crash |
|-----------------------|-----------------------|--------------------------------|
| Brand: Xiaomi | Version: Android 12 | Date: Nov 14, 2022, 1:42:56 AM |
| Model: Mi 10T Pro | Orientation: Portrait | App version: 1.0 (1) |
| Orientation: Portrait | Rooted: No | |
| RAM free: 2.94 GB | | |
| Disk free: 146.39 GB | | |

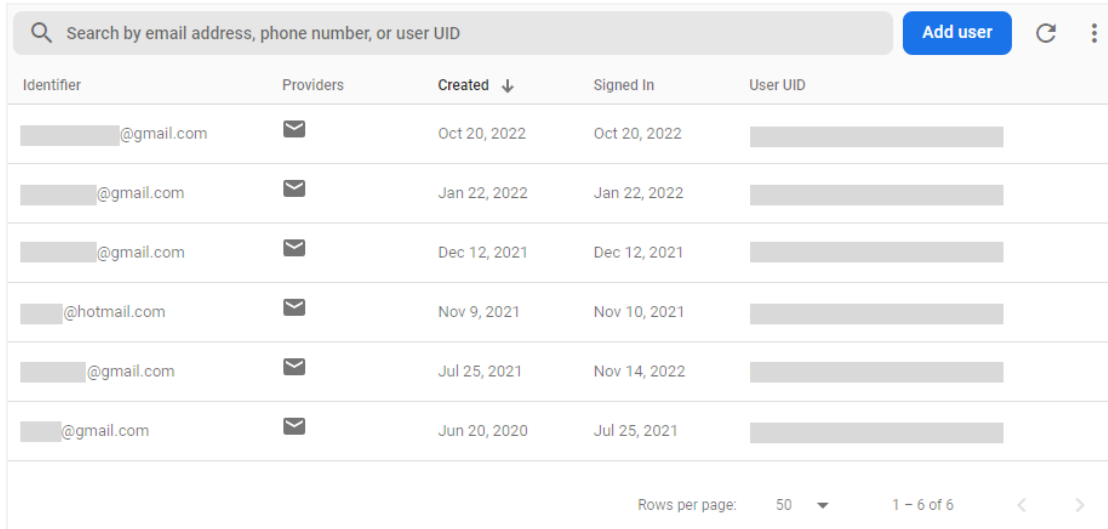
Εικόνα 112. Επισκόπηση στοιχείων σχετικών με το υπό επεξεργασία σφάλμα

6.2.2.2. Firebase Authentication¹⁸

Η συγκεκριμένη υπηρεσία διευκολύνει την εγγραφή και σύνδεση των χρηστών στην εφαρμογή. Η επιβεβαίωση των στοιχείων μπορεί να γίνει με πλήθος τρόπων, όπως η χρήση κινητού τηλεφώνου, διεύθυνσης ηλεκτρονικού ταχυδρομείου, ή λογαριασμού Facebook και πολλών ακόμα.

Για τις ανάγκες της παρούσας εφαρμογής χρησιμοποιήθηκε η διεύθυνση ηλεκτρονικού ταχυδρομείου ως αναγνωριστικό χρήστη. Παρακάτω φαίνεται η λίστα των ήδη υπάρχοντων χρηστών στην εφαρμογή:





| Identifier | Providers | Created ↓ | Signed In | User UID |
|------------------------|-----------|--------------|--------------|------------|
| ██████████@gmail.com | ✉ | Oct 20, 2022 | Oct 20, 2022 | ██████████ |
| ██████████@gmail.com | ✉ | Jan 22, 2022 | Jan 22, 2022 | ██████████ |
| ██████████@gmail.com | ✉ | Dec 12, 2021 | Dec 12, 2021 | ██████████ |
| ██████████@hotmail.com | ✉ | Nov 9, 2021 | Nov 10, 2021 | ██████████ |
| ██████████@gmail.com | ✉ | Jul 25, 2021 | Nov 14, 2022 | ██████████ |
| ██████████@gmail.com | ✉ | Jun 20, 2020 | Jul 25, 2021 | ██████████ |

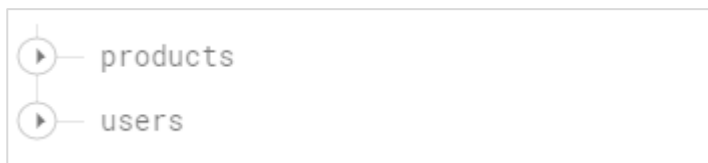
Εικόνα 113. Αυθεντικοποίηση χρηστών

Η λίστα αυτή περιλαμβάνει στοιχεία όπως η ημερομηνία δημιουργίας του λογαριασμού και την ημερομηνία τελευταίας σύνδεσης, στοιχείο χρήσιμο για την αναγνώριση της επισκεψιμότητας στην εφαρμογή μας. Από εδώ, ο προγραμματιστής μπορεί να δημιουργήσει έναν νέο χρήστη, να επαναφέρει τον κωδικό ενός ήδη υπάρχοντος στέλνοντάς του μήνυμα ηλ. ταχυδρομείου με αυτόν τον σκοπό, ή να τον διαγράψει εξολοκλήρου.

6.2.2.3. Firebase Realtime Database¹⁹

Για τη δημιουργία της βάσης δεδομένων και την παρακολούθηση των αλλαγών που πραγματοποιούνται σε αυτήν σε πραγματικό χρόνο, χρησιμοποιήθηκε η παρούσα υπηρεσία. Η υπηρεσία Realtime Database αφορά σε μία NoSQL βάση δεδομένων, δηλαδή δεν περιλαμβάνει πίνακες ή εγγραφές, αλλά αποθηκεύει τις πληροφορίες που δέχεται σε JSON μορφή. Η βάση αυτή βρίσκεται στο cloud της Firebase.

Για την υλοποίηση της παρούσας εφαρμογής δημιουργήθηκαν δύο βασικοί κόμβοι για την αποθήκευση της πληροφορίας, όπως φαίνεται παρακάτω:



Εικόνα 114. Realtime Database - Βασικοί κόμβοι

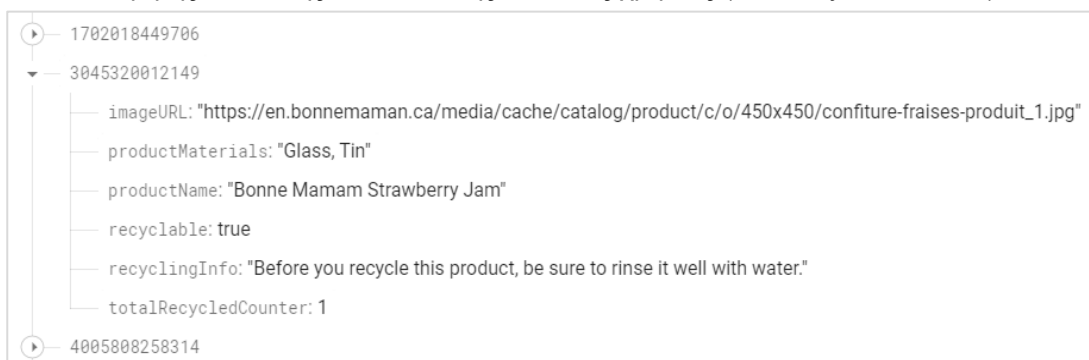
Εκ των οποίων ο κόμβος «products» περιέχει τη λίστα των προϊόντων, οργανωμένα με βάση τον κωδικό barcode τους.



Εικόνα 115. Realtime Database - Κόμβος Προϊόντων

Στο εσωτερικό βρίσκονται οι επιμέρους πληροφορίες του κάθε προϊόντος, οι οποίες αποτελούνται από τα ακόλουθα χαρακτηριστικά:

- Σύνδεσμος εικόνας προϊόντος (*imageURL*),
- Υλικά από τα οποία αποτελείται το προϊόν (*productMaterials*),
- Όνομα προϊόντος (*productName*),
- Ανακυκλωσιμότητα ή μη (*recyclable*),
- Πληροφορίες/ Συμβουλές για την ανακύκλωση ή μη του προϊόντος (*recyclingInfo*),
- Μετρητής συνολικής ανακύκλωσης από τους χρήστες (*totalRecycledCounter*).

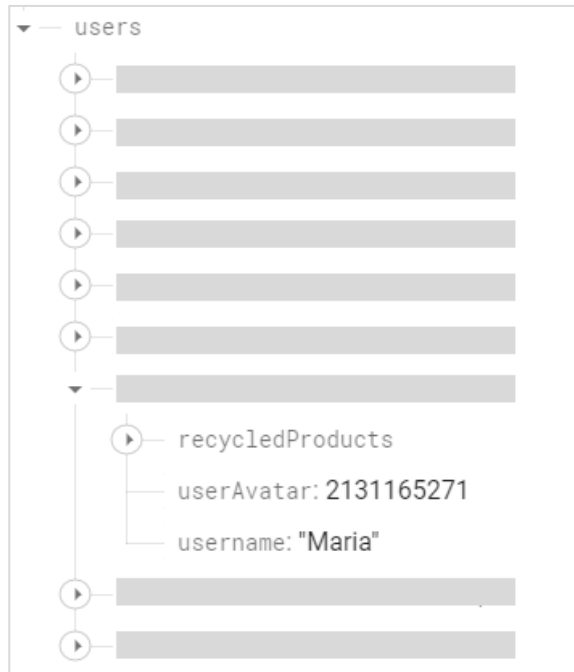


Εικόνα 116. Realtime Database - Κόμβος Συγκεκριμένου Προϊόντος

Ο μετρητής ξεκινά από το 0 και ανανεώνεται αφότου κάποιος χρήστης ενημερώσει ότι έχει ανακυκλώσει το εν λόγω προϊόν.

Ο δεύτερος κόμβος αφορά στους χρήστες της εφαρμογής και κάθε υπο-κόμβος περιέχει την πληροφορία του μοναδικού UID (αναγνωριστικό χρήστη) που δημιουργείται κατά την εγγραφή ενός χρήστη στο σύστημα της Firebase μέσω της υπηρεσίας της Firebase Authentication για να τους διαχωρίσει. Το κάθε αναγνωριστικό χρήστη, περιέχει με τη σειρά του υπο-κόμβους που αφορούν στο επιλεγμένο άβαταρ χρήστη, στο ορισμένο όνομα χρήστη και στο σύνολο των ανακυκλωμένων προϊόντων του.





Εικόνα 117. Realtime Database - Κόμβος χρήστη

Η τιμή του επιλεγμένου άβαταρ χρήστη (`userAvatar`) αναφέρεται σε αναγνωριστικό εικονιδίου που βρίσκεται τοπικά στην εφαρμογή, ενώ το όνομα χρήστη έχει ως προεπιλογή το λεκτικό «Username». Στη λίστα ανακυκλωμένων προϊόντων του κάθε χρήστη (`recycledProducts`) τα προϊόντα αυτά οργανώνονται με αναγνωριστικό έναν ακέραιο αριθμό αυξητικό κάθε φορά κατά ένα. Στο εσωτερικό αυτών βρίσκονται οι εξής τιμές: ο barcode κωδικός προϊόντος (`barcodeValue`) και η χρονοσήμανση της ανακύκλωσής του (`recycledTimestamp`).



Εικόνα 118. Realtime Database – Κόμβος Ανακυκλωμένων Προϊόντων

6.2.2.4. Firebase ML Kit²⁰

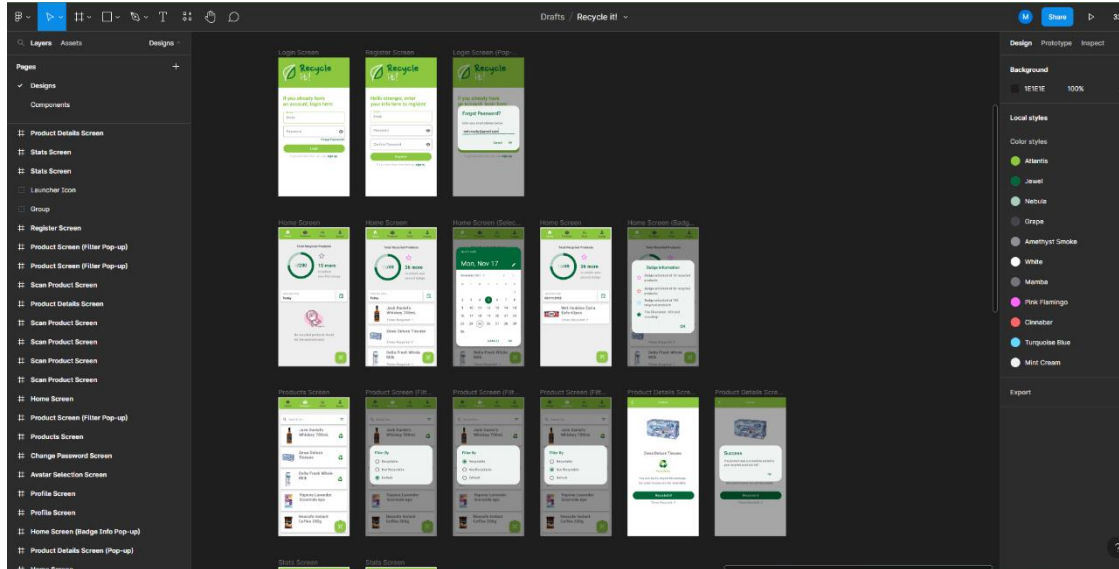
Για την αναγνώριση κωδικών barcode χρησιμοποιήθηκε το Barcode Scanning της υπηρεσίας ML Kit της Firebase. Με αυτό είναι δυνατή η ανάγνωση κωδικοποιημένων δεδομένων για τις περισσότερες περιπτώσεις τύπων barcode, χωρίς την ανάγκη σύνδεσης με το διαδίκτυο και χωρίς την ανάγκη για καθορισμένο προσανατολισμό στην τοποθέτησή τους. Οι μορφές των κωδικών barcode που μπορούν να αναγνωρισθούν χωρίζονται στις κατηγορίες των γραμμικών, που περιλαμβάνουν τα Codabar, Code 39, Code 93, EAN-8, EAN-13, ITF, UPC-A, UPC-E, και των διδιάστατων (2D), που περιέχουν τα Aztec, Data Matrix, PDF417 και QR Code²¹.

6.3.3. Figma²²

Το Figma είναι ένα διαδικτυακό εργαλείο, το οποίο επίσης παρέχεται σε εφαρμογές για Windows και macOS. Το σημαντικότερο πλεονέκτημα του εργαλείου αυτού είναι πως όλες οι παροχές του μπορούν να χρησιμοποιηθούν διαδικτυακά, καθιστώντας εφικτή την ταυτόχρονη ή ασύγχρονη συνεργασία μεταξύ μίας ομάδας ατόμων. Αυτό γίνεται δυνατό καθώς η κάθε εργασία που επιτελείται σε ένα αρχείο, αποθηκεύεται αυτόματα στο cloud, ενώ διατηρείται μία λίστα με όλες τις αλλαγές που έχουν πραγματοποιηθεί ανά διαστήματα χρόνου (*version history*), καθιστώντας το εύκολο προς τον χρήστη να επιστρέψει σε παλαιότερη έκδοση, αν το επιθυμεί. Επιπλέον, τα άτομα που εμπλέκονται στο εν λόγω αρχείο, ή και άτομα εκτός εκείνων τα οποία έχουν πρόσβαση σε αυτό, έχουν τη δυνατότητα παροχής ανατροφοδότησης, χάρη στο τμήμα των σχολίων του Figma. Τα σχόλια αυτά παραμένουν, εκτός βέβαια εάν ο δημιουργός επιθυμήσει να τα σβήσει, ώστε να ανατρέξει σε αυτά αν το κρίνει απαραίτητο.

Ένα ακόμα πλεονέκτημα του Figma είναι πως αποτελεί μία πλατφόρμα την οποία οι χρήστες μπορούν να εκμεταλλευθούν για άντληση ιδεών, ή ενσωμάτωσή τους στο έργο τους. Οι ιδέες αυτές είναι σχετικές με όλες τις δυνατότητες που παρέχει το εν λόγω εργαλείο, όπως η σχεδίαση διεπαφής χρήστη (UI), η δημιουργία προσχέδιων (π.χ. για μία εφαρμογή), η δημιουργία εικονιδίων και η δημιουργία διαγραμμάτων ροής ή διαγραμμάτων UML.

Παραδείγματα της χρήσης του Figma στην παρούσα εφαρμογή αποτελούν η λίστα οθονών στην Ενότητα *Εγχειρίδιο Χρήστη*, καθώς και η πληθώρα των UML διαγραμμάτων που παρατίθενται. Παρακάτω φαίνεται το αρχείο που περιέχει τις οθόνες της εφαρμογής:



Εικόνα 119. Στιγμιότυπο οθόνης από το Figma

6.3.4. LottieFiles²³

Πρόκειται για μία πλατφόρμα η οποία περιέχει ένα πλήθος *animations*, έτοιμα για χρήση. Περιλαμβάνει αρχεία τύπου *.json* τα οποία διατηρούν την ποιότητά τους σε οποιοδήποτε μέγεθος και είναι αρκετά «ελαφριά» ώστε να μην επιβαρύνουν την εφαρμογή.

Για τη χρήση ενός αρχείου *Lottie*, χρειάζεται η προσθήκη της παρακάτω *dependency* στο project:

```
// Lottie Animation
implementation 'com.airbnb.android:lottie:3.4.0'
```

Εικόνα 120. Προσθήκη *dependency* στο project για τη χρήση *Lottie* αρχείων

Στη συνέχεια, κάνουμε τις απαραίτητες προσθήκες στο αρχείο XML που μας ενδιαφέρει:




```
<com.airbnb.lottie.LottieAnimationView
    android:id="@+id/progress_animation"
    android:layout_width="200dp"
    android:layout_height="150dp"
    app:lottie_autoPlay="true"
    app:lottie_fileName="progress_animation.json"
    app:lottie_loop="true"
    android:visibility="gone"
    app:layout_constraintTop_toBottomOf="@id/btn_sign_up"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintVertical_bias="0" />
```

Εικόνα 121. Στιγμιότυπο χρήσης Lottie αρχείο στην εφαρμογή

Για την εμφάνισή του στην αντίστοιχη οθόνη επηρεάζουμε την τιμή *visibility*, ανάλογα με την περίπτωση.

6.4. Βιβλιοθήκες

6.4.1. Android Jetpack²⁴

Το Jetpack αποτελεί μία συλλογή βιβλιοθηκών, με σκοπό την υποβοήθηση των προγραμματιστών να ακολουθήσουν τις βέλτιστες πρακτικές, τη μείωση επαναλαμβανόμενου κώδικα, καθώς και την παροχή κώδικα που ανταποκρίνεται σε πλήθος συσκευών και εκδόσεων Android.

6.4.1.1. View Binding²⁵

Πρόκειται για μία λειτουργία, η οποία, όταν ενεργοποιηθεί, προσφέρει τη δυνατότητα ευκολότερης πρόσβασης στα *XML layout* αρχεία και τα συστατικά τους. Αυτό καθίσταται δυνατό, διότι για κάθε *layout* αρχείο που δημιουργείται, παράγεται και μία *binding* τάξη, η οποία παρέχει μία άμεση σύνδεση/ αναφορά σε στοιχεία που περιέχονται σε αυτό και έχουν κάποιο αναγνωριστικό (*id*). Με τον τρόπο αυτό, μπορεί να αντικατασταθεί η χρήση της μεθόδου *findViewById*.

Η επιλογή της συγκεκριμένης λειτουργίας έγινε λόγω των πλεονεκτημάτων που προσφέρει έναντι της μεθόδου *findViewById*, τα οποία συνίστανται στην αποφυγή σφαλμάτων τύπου *NullPointerException*, που προκύπτουν κατά την εισαγωγή *ID* που δεν ανήκει στο εκάστοτε *layout* αρχείο, καθώς και *ClassCastException*, που εμφανίζονται κατά τη λάθος αντιστοίχιση ενός στοιχείου με τον τύπο στον οποίο ανήκει (π.χ. *Button*).

6.4.1.2. Fragments²⁶

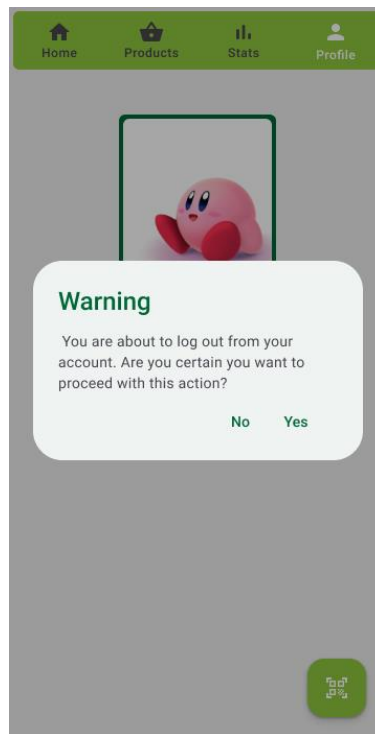
Τα *Fragments* χρησιμοποιούνται για την τμηματοποίηση της διεπαφής του χρήστη σε μικρότερα, πιο διαχειρίσιμα κομμάτια. Κατέχουν το δικό τους *lifecycle*, αλλά εξαρτώνται αποκλειστικά από κάποιον ξενιστή, ο οποίος πιο συχνά είναι μία *Activity*. Με τη χρήση τους ενισχύεται η επαναχρησιμοποίηση κώδικα, αλλά και η υποστήριξη πολλαπλών οθονών, διαφορετικών



διαστάσεων. Εκτός αυτού καθίστανται αναγκαία για ορισμένες βιβλιοθήκες της *Android Jetpack*, όπως η *Navigation* και *Bottom Navigation View*, οι οποίες αναπτύσσονται σε επόμενες ενότητες.

6.4.1.3. Dialog Fragments²⁷

Τα Dialog Fragments είναι αναδυόμενα παράθυρα, τα οποία προσφέρουν ανατροφοδότηση στον χρήστη, ή ζητούν κάποια ενέργεια από εκείνον. Αναφέρεται πως η τάξη Dialog Fragment παρέχει σωστότερη αντιμετώπιση των γεγονότων lifecycle, όπως για παράδειγμα όταν ο χρήστης πατήσει το κουμπί «Πίσω» ή αν η οθόνη περιστραφεί. Έτσι, για τις ανάγκες της παρούσας εφαρμογής έγινε χρήση του σε συνδυασμό με ειδικό σχεδιασμό (*custom layout*) και τις κατευθυντήριες οδηγίες της Material. Παρακάτω παρατίθεται ο κώδικας για την υλοποίησή ενός από αυτά που χρησιμοποιήθηκαν για την εφαρμογή.



Εικόνα 122. Πρότυπο αναδυομένου παραθύρου για την επιβεβαίωση ενέργειας

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <androidx.cardview.widget.CardView
        android:layout_width="0dp"
        app:layout_constraintWidth_percent="0.85"
        android:layout_height="wrap_content"
        app:cardCornerRadius="28dp"
        app:cardBackgroundColor="@color/mint_cream"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintBottom_toBottomOf="parent">

        <androidx.constraintlayout.widget.ConstraintLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent">

            <androidx.constraintlayout.widget.Guideline
                android:id="@+id/guideline_start"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:orientation="vertical"
                app:layout_constraintGuide_percent="0.05"/>

            <androidx.constraintlayout.widget.Guideline
                android:id="@+id/guideline_end"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:orientation="vertical"
                app:layout_constraintGuide_percent="0.95"/>

            <TextView
                android:id="@+id/title"
```

Εικόνα 123. Κώδικας XML για τη δημιουργία ειδικού layout αναδυομένου παραθύρου - Στιγμιότυπο Οθόνης 1



```
<TextView
    android:id="@+id/title"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="Title"
    android:textColor="@color/jewel"
    android:textAppearance="@style/robotofontStyleBoldLarge"
    android:paddingTop="20dp"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintStart_toStartOf="@id/guideline_start"
    app:layout_constraintEnd_toEndOf="@id/guideline_end"/>

<TextView
    android:id="@+id/message"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:paddingVertical="10dp"
    android:text="Message"
    android:textAppearance="@style/robotofontStyleRegularSmall"
    android:textColor="@color/grape"
    app:layout_constraintVertical_bias="0"
    app:layout_constraintTop_toBottomOf="@id/title"
    app:layout_constraintStart_toStartOf="@id/guideline_start"
    app:layout_constraintEnd_toEndOf="@id/guideline_end"/>
```

Εικόνα 124. Κώδικας XML για τη δημιουργία ειδικού layout αναδυομένου παραθύρου - Στιγμιότυπο Οθόνης 2

```

<Button
    android:id="@+id/btn_action_1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Action 1"
    android:textColor="@color/jewel"
    android:textAppearance="@style/robotofontStyleMediumSmall"
    style="@style/Widget.Material3.Button.TextButton"
    app:layout_constraintTop_toBottomOf="@id/message"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintHorizontal_bias="1"
    android:layout_marginHorizontal="10dp"
    android:layout_marginBottom="10dp"/>

<Button
    android:id="@+id/btn_action_2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Action 2"
    android:textColor="@color/jewel"
    android:textAppearance="@style/robotofontStyleMediumSmall"
    style="@style/Widget.Material3.Button.TextButton"
    app:layout_constraintTop_toBottomOf="@id/message"
    app:layout_constraintEnd_toStartOf="@id/btn_action_1"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintHorizontal_bias="1"
    android:layout_marginBottom="10dp"/>
</androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>

</androidx.constraintlayout.widget.ConstraintLayout>

```

Εικόνα 125. Κώδικας XML για τη δημιουργία ειδικού layout αναδυομένου παραθύρου - Στιγμιότυπο Οθόνης 3

Το αναδυόμενο παράθυρο *Custom Confirmatory Dialog Fragment* πρόκειται για ένα παράθυρο γενικευμένης χρήσης κατά το οποίο ο χρήστης καλείται να συμπληρώσει τις εξής πληροφορίες:

- Τίτλος (*title*): Υποχρεωτικό πεδίο. Εμφανίζεται στο πάνω μέρος του παραθύρου στα αριστερά.
- Μήνυμα (*message*): Υποχρεωτικό πεδίο. Καταλαμβάνει το εσωτερικό μέρος του παραθύρου και πλαισιώνεται από τον τίτλο και τα κουμπιά.



- Ενέργεια 1^η (*action1*): Υποχρεωτικό πεδίο. Αναφέρεται στο κείμενο που θα προβάλλεται στο πρώτο κουμπί του παραθύρου που βρίσκεται στο κάτω μέρος και δεξιά.
- Ενέργεια 2^η (*action2*): Υποχρεωτικό πεδίο. Αντιπροσωπεύει το κείμενο του δεύτερου κατά σειρά κουμπιού από δεξιά.
- Ακυρώσιμο (*cancelable*): Προαιρετικό πεδίο. Αφορά στην ικανότητα του παραθύρου να «εξαφανίζεται» εάν ο χρήστης ακουμπήσει σε οποιοδήποτε σημείο της οθόνης εκτός του παραθύρου ή πατήσει το κουμπί «Πίσω». Από προεπιλογή το παράθυρο ορίζεται ως ακυρώσιμο.
- Listener: Υποχρεωτικό πεδίο. Χάρη σε αυτό το πεδίο παρέχεται ανατροφοδότηση σχετικά με την επιλογή του χρήστη. Πιο συγκεκριμένα, στην περίπτωση του παραθύρου που εμφανίζεται πριν την Αποσύνδεση του χρήστη, θα έχουμε τη δυνατότητα να γνωρίζουμε αν τελικά ο χρήστης επέλεξε το κουμπί «Ναι», ή το κουμπί «Όχι».

Για τη δημιουργία του παραθύρου γίνεται *override* της μεθόδου *onCreateDialog()* της τάξης *Dialog*. Σε αυτή ορίζεται το ειδικό *layout* που έχει δημιουργηθεί για αυτήν ακριβώς την περίπτωση. Η υλοποίησή του πραγματοποιείται με τη χρήση του *MaterialAlertDialogBuilder*. Στην συνέχεια ορίζονται τα βασικά στοιχεία του παραθύρου, όπως αναφέρθηκαν νωρίτερα (τίτλος, μήνυμα κ.λπ.), βάσει των πεδίων που συμπληρώνονται κατά την κλήση του παραθύρου.

```

class CustomConfirmatoryDialogFragment (
    private val title: String,
    private val message: String,
    private val action1: String,
    private val action2: String,
    private val cancelable: Boolean = true,
    private val listener: Listener
): DialogFragment() {
    private val TAG = CustomConfirmatoryDialogFragment::class.java.name
    private var _binding: CustomConfirmatoryDialogBinding? = null
    private val binding get() = _binding!!
    private var isShowing = false

    interface Listener {
        fun btnAction1()
        fun btnAction2()
    }

    override fun onCreateDialog(savedInstanceState: Bundle?): Dialog {
        Log.i(TAG, msg: "onCreateDialog")

        _binding = CustomConfirmatoryDialogBinding.inflate(requireActivity().layoutInflater)
        val builder = MaterialAlertDialogBuilder(this.requireActivity())
        builder.setView(binding.root)

        binding.title.text = title
        binding.message.text = message

        binding.btnAction1.text = action1
        binding.btnAction2.text = action2

        binding.btnAction1.setOnClickListener { it: View!
            dismiss()
            listener.btnAction1()
        }

        binding.btnAction2.setOnClickListener { it: View!
            dismiss()
            listener.btnAction2()
        }
    }
}

```

Εικόνα 126. Κώδικας για την υλοποίηση αναδυόμενου παραθύρου με τη χρήση του Dialog Fragment - Στιγμιότυπο Οθόνης 1

Κατά τη διαδικασία δημιουργίας του παραθύρου - *onCreateDialog()* – ορίζεται η τοποθέτηση του παραθύρου στην οθόνη, που στη συγκεκριμένη περίπτωση είναι στο κέντρο και η δυνατότητα του παραθύρου να «εξαφανίζεται», όπως αναφέρθηκε προηγουμένως (*isCancelable*).

```
        return builder.create()
    }

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        val window: Window? = dialog?.window
        val wlp: WindowManager.LayoutParams? = window?.attributes
        wlp?.gravity = Gravity.CENTER
        window?.attributes = wlp
        window?.setBackgroundDrawable(ColorDrawable(Color.TRANSPARENT))

        isCancelable = cancelable

        return super.onCreateView(inflater, container, savedInstanceState)
    }

    override fun show(manager: FragmentManager, tag: String?) {
        // check if it is already showing (to avoid crash)
        if (isShowing) {
            Log.i(TAG, msg: "Dialog is already showing")
            return
        }
        isShowing = true
        super.show(manager, tag)
    }

    override fun onDismiss(dialog: DialogInterface) {
        Log.i(TAG, msg: "dismiss()")
        isShowing = false
        super.dismiss()
    }
}
```

Εικόνα 127. Κώδικας για την υλοποίηση αναδυόμενου παραθύρου με τη χρήση του Dialog Fragment - Στιγμιότυπο Οθόνης 2




```

override fun onDestroyView() {
    Log.i(TAG, msg: "onDestroy()")
    isShowing = false
    _binding = null
    super.onDestroyView()
}
}

```

Εικόνα 128. Κώδικας για την υλοποίηση αναδυόμενου παραθύρου με τη χρήση του Dialog Fragment - Στιγμιότυπο Οθόνης 3

Παρακάτω παρατίθεται τμήμα του κώδικα που περιέχει παράδειγμα κλήσης του CustomConfirmatoryDialogFragment που παρουσιάστηκε παραπάνω.

```

binding.btnLogout.setOnClickListener { it: View?
    Log.i(TAG, msg: "btnLogout was clicked.")

    // show confirmatory pop-up
    if (confirmatoryDialog != null) {
        confirmatoryDialog!!.dismiss()
    }
    confirmatoryDialog = CustomConfirmatoryDialogFragment(
        title = getString(R.string.warning),
        message = getString(R.string.log_out_confirmation),
        action1 = getString(R.string.yes),
        action2 = getString(R.string.no),
        cancelable = false,
        listener = object : CustomConfirmatoryDialogFragment.Listener {
            override fun btnAction1() {
                Log.i(TAG, msg: "btnAction1 was clicked.")
                // if yes, then go back to LoginActivity
                FirebaseAuth.getInstance().signOut()
                val intent = Intent(activity, LoginActivity::class.java)
                intent.flags = Intent.FLAG_ACTIVITY_NEW_TASK or Intent.FLAG_ACTIVITY_CLEAR_TASK
                startActivity(intent)
                requireActivity().finish()
            }

            override fun btnAction2() {
                Log.i(TAG, msg: "btnAction2 was clicked.")
            }
        }
    )
    confirmatoryDialog!!.show(requireActivity().supportFragmentManager, tag: "CustomConfirmatoryDialogFragment")
}
}

```

Εικόνα 129. Στιγμιότυπο οθόνης για την κλήση αναδυόμενου παραθύρου

Με το πάτημα του κουμπιού για την αποσύνδεση του χρήστη από την εφαρμογή, προβάλλεται ένα παράθυρο επιβεβαίωσης της ενέργειάς του. Κατά την κλήση του, και πριν την εμφάνισή του, ορίζονται τα απαραίτητα πεδία του, όπως ο *listener*, όπου όπως φαίνεται στην παραπάνω εικόνα, είναι υπεύθυνος για τις ενέργειες που θα πραγματοποιηθούν κατά την επιλογή οποιωνδήποτε από την δύο ενέργειες που παρέχονται στον χρήστη. Στη συγκεκριμένη περίπτωση, κατά την επιλογή της δεύτερης ενέργειας, η οποία ισοδυναμεί με την αρνητική απάντηση, δεν γίνεται κάποια αλλαγή, ενώ με την πρώτη (θετική απάντηση) πραγματοποιείται αποσύνδεση από το σύστημα *Authentication* της *Firebase* και επιστροφή στην οθόνη σύνδεσης (*LoginActivity*).



6.4.1.4. Navigation²⁸

Το στοιχείο *Navigation* της βιβλιοθήκης *Jetpack* αποτελεί σημαντικό και εξαιρετικά χρήσιμο εργαλείο για την υλοποίηση της πλοήγησης του χρήστη μεταξύ των οθονών της εφαρμογής. Παρέχει τη δυνατότητα ελέγχου των πλοηγήσεων αυτών, ενώ με την οργάνωσή τους καθορίζει μία συνεπή και επομένως πιο ευχάριστη εμπειρία στον χρήστη. Επιπρόσθετα, υποστηρίζει τα *View Models*, τα οποία θα αναπτυχθούν εκτενέστερα σε επόμενη ενότητα, καθώς επίσης και τα *Safe Args* (*Safe Arguments*). Πρόκειται για τη μεταφορά δεδομένων από τον έναν προορισμό στον επόμενο, διαφυλάσσοντας όμως την ασφάλεια όσον αφορά στον τύπο των δεδομένων.

```
<fragment
    android:id="@+id/productDetailsFragment"
    android:label="Product Materials"
    tools:layout="@layout/fragment_product_details"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.ProductDetailsFragment">
    <argument
        android:name="product_selected"
        app:argType="string"
        android:defaultValue="" />
</fragment>
```

Εικόνα 130. Στιγμιότυπο οθόνης - Ορισμός δεδομένου προς μεταφορά κατά την πλοήγηση μεταξύ προορισμών

Στο παραπάνω παράδειγμα φαίνεται ο ορισμός ενός τέτοιου δεδομένου, τύπου αλφαριθμητικού (*string*), στο εσωτερικό του προορισμού ονόματι *productDetailsFragment*. Χάρη σε αυτό διασφαλίζεται πως κατά την πλοήγηση από οποιονδήποτε προορισμό προς τον παρόντα, θα μεταφέρεται η πληροφορία του προϊόντος που έχει επιλεγεί.

Το *Navigation* περιλαμβάνει τρία βασικά χαρακτηριστικά, τα οποία θα παρουσιαστούν με αναφορές σε αντίστοιχα κομμάτια υλοποίησης στην παρούσα εργασία.

- **Navigation Graph:** Αφορά σε ένα αρχείο *XML*, που περιλαμβάνει το σύνολο των πιθανών προορισμών, καθώς και των πλοηγήσεων από και προς σε αυτούς. Κάθε αρχείο *navigation graph* περιέχει απαραίτητως έναν αρχικό προορισμό (*start destination*), ο οποίος αποτελεί την πρώτη οθόνη που θα εμφανιστεί.

Παρακάτω προβάλλονται μία σειρά στιγμιότυπων οθόνης από το αρχείο *Navigation Graph* που υλοποιήθηκε στην εφαρμογή. Στο άνω μέρος ορίζεται ο αρχικός προορισμός, ο οποίος τυχαίνει να είναι η Αρχική Οθόνη (*homeFragment*) και στη συνέχεια το αναγνωριστικό του αρχείου. Στην πορεία ορίζεται ο κάθε ξεχωριστός προορισμός (*homeFragment*, *profileFragment*, κ.λπ.), στο εσωτερικό των οποίων προσδιορίζονται οι ενέργειες πλοήγησης από τον τωρινό προορισμό προς κάποιον επόμενο. Για τον προσδιορισμό των δεύτερων απαιτείται ένα αναγνωριστικό (π.χ. *action_homeFragment_to_scanFragment*), καθώς επίσης και ο τελικός ορισμός (π.χ. *scanFragment*). Το τελευταίο στοιχείο είναι αναγκαίο να αναφέρεται σε υπαρκτό στοιχείο στο αρχείο.



```
<?xml version="1.0" encoding="utf-8"?>
<navigation xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  app:startDestination="@id/homeFragment"
  android:id="@+id/main_nav_graph">

  <fragment
    android:id="@+id/homeFragment"
    android:label="Home"
    tools:layout="@layout/fragment_home"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.HomeFragment"
    <action
      android:id="@+id/action_homeFragment_to_productDetailsFragment"
      app:destination="@id/productDetailsFragment" />
    <action
      android:id="@+id/action_homeFragment_to_scanFragment"
      app:destination="@id/scanFragment" />
    </fragment>

  <fragment
    android:id="@+id/profileFragment"
    android:label="Profile"
    tools:layout="@layout/fragment_profile"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.ProfileFragment"
    <action
      android:id="@+id/action_profileFragment_to_chooseAvatarFragment"
      app:destination="@id/chooseAvatarFragment" />
    <action
      android:id="@+id/action_profileFragment_to_scanFragment"
      app:destination="@id/scanFragment" />
    <action
      android:id="@+id/action_profileFragment_to_changePasswordFragment"
      app:destination="@id/changePasswordFragment" />
    </fragment>
</navigation>
```

Εικόνα 131. Στιγμιότυπο οθόνης 1 - Αρχείο Navigation Graph



```
<fragment
    android:id="@+id/chooseAvatarFragment"
    android:label="Choose an Avatar"
    tools:layout="@layout/fragment_choose_avatar"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.ChooseAvatarFragm

<fragment
    android:id="@+id/changePasswordFragment"
    android:label="Change Password"
    tools:layout="@layout/fragment_change_password"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.ChangePasswordFra

<fragment
    android:id="@+id/productDisplayFragment"
    android:label="Products"
    tools:layout="@layout/fragment_product_display"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.ProductDisplayFra
    <action
        android:id="@+id/action_productDisplayFragment_to_productDetails
        app:destination="@id/productDetailsFragment" />
    <action
        android:id="@+id/action_productDisplayFragment_to_scanFragment"
        app:destination="@id/scanFragment" />
</fragment>

<fragment
    android:id="@+id/productDetailsFragment"
    android:label="Product Materials"
    tools:layout="@layout/fragment_product_details"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.ProductDetailsFra
    <argument
        android:name="product_selected"
        app:argType="string"
        android:defaultValue="" />
</fragment>
```

Εικόνα 132. Στιγμιότυπο οθόνης 2 - Αρχείο Navigation Graph

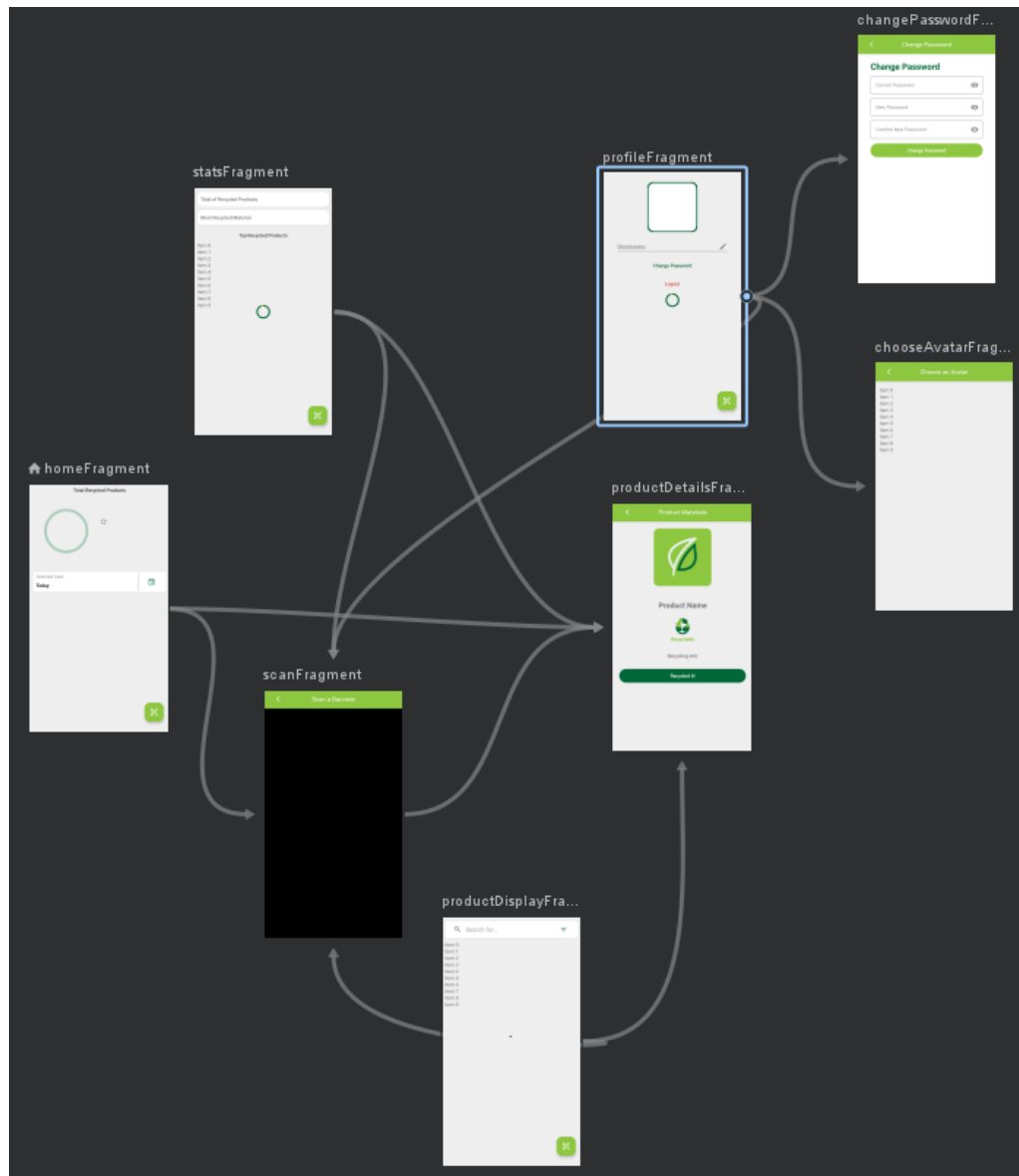


```
<fragment
    android:id="@+id/statsFragment"
    android:label="Statistics"
    tools:layout="@layout/fragment_stats"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.StatsFrag
<action
    android:id="@+id/action_statsFragment_to_scanFragment"
    app:destination="@id/scanFragment" />
<action
    android:id="@+id/action_statsFragment_to_productDetailsFr
    app:destination="@id/productDetailsFragment" />
</fragment>

<fragment
    android:id="@+id/scanFragment"
    android:label="Scan a Barcode"
    tools:layout="@layout/fragment_scan"
    android:name="gr.mlygeroudi.recycleit.ui.fragments.ScanFragm
<action
    android:id="@+id/action_scanFragment_to_productDetailsFr
    app:destination="@id/productDetailsFragment" />
</fragment>
</navigation>
```

Εικόνα 133. Στιγμιότυπο οθόνης 3 - Αρχείο Navigation Graph

Στη συνέχεια παρατίθεται μία οπτική παρουσίαση των ορισμένων προορισμών και των πλοηγήσεων μεταξύ τους.



Εικόνα 134. Οπτική παρουσίαση του Navigation Graph - Στιγμιότυπο οθόνης

- **NavHost:** Πρόκειται για ένα στοιχείο, το οποίο προβάλλει τους προορισμούς που ορίστηκαν στο *Navigation Graph*.

Για την προβολή τους, χρειάζεται να προστεθεί ο ακόλουθος κώδικας στην *Activity* που θα στεγάσει τη λίστα των προορισμών (*Fragments*), τα οποία περιέχονται στο αρχείο *Navigation Graph*. Στη συγκεκριμένη περίπτωση το αρχείο αυτό ονομάζεται *main_nav_graph*, το οποίο πρέπει απαραίτητως να ορισθεί εδώ.

```
<androidx.fragment.app.FragmentContainerView
    android:id="@+id/nav_host_fragment_container"
    android:name="androidx.navigation.fragment.NavHostFragment"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    app:navGraph="@navigation/main_nav_graph"
    app:defaultNavHost="true"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@id/bottom_nav"
    app:layout_constraintBottom_toBottomOf="parent"/>
```

Εικόνα 135. Ενσωμάτωση NavHost σε μία Activity - Στιγμιότυπο οθόνης

Για την πρόσβαση στον *NavHost* είναι απαραίτητο να χρησιμοποιηθεί ο ακόλουθος κώδικας.

```
val navHostFragment: NavHostFragment = supportFragmentManager
    .findFragmentById(R.id.nav_host_fragment_container) as NavHostFragment
```

Εικόνα 136. Πρόσβαση στον NavHost – Στιγμιότυπο οθόνης

- **NavController:** Το συγκεκριμένο στοιχείο αναλαμβάνει την πλοήγηση μεταξύ των προορισμών, που έχουν ορισθεί στο *Navigation Graph* αρχείο, στο εσωτερικό του *NavHost*.

Αυτό είναι πιο κατανοητό με το παρακάτω τμήμα του κώδικα.

```
val navController = navHostFragment.findNavController()
```

Εικόνα 137. Ορισμός NavController - Στιγμιότυπο οθόνης

Για τον ορισμό του *NavController* είναι αναγκαία η πρόσβαση στον *NavHost*.

Παρακάτω ακολουθεί παράδειγμα της χρήσης του *NavController* για την πλοήγηση από τον προορισμό Οθόνη Προφίλ (*Profile Fragment*) προς την Οθόνη Σάρωσης (*Scan Fragment*).

```
// navigate to ScanFragment
val action = ProfileFragmentDirections.actionProfileFragmentToScanFragment()
findNavController().navigate(action)
```

Εικόνα 138. Τμήμα κώδικα για την πλοήγηση μεταξύ προορισμών με τη χρήση του NavController – Στιγμιότυπο οθόνης 1

Παρακάτω ακολουθεί παράδειγμα της χρήσης του *NavController* για την πλοήγηση από τον προορισμό Αρχική Οθόνη (*Home Fragment*) προς την Οθόνη Προβολής Στοιχείων Προϊόντος (*Product Details Fragment*), με την επιπρόσθετη μεταφορά του δεδομένου «*productSelected*», το οποίο αποτελεί *argument* που ορίστηκε στο αρχείο *Navigation Graph* και αφορά τον κωδικό barcode προϊόντος που επιλέχθηκε. Έτσι, μόλις ο χρήστης μεταβεί στη δεύτερη οθόνη, θα προβληθούν τα στοιχεία του προϊόντος με τον αντίστοιχο κωδικό.



```
val action = HomeFragmentDirections.actionHomeFragmentToProductDetailsFragment(  
    productSelected = barcodeSelected  
)  
findNavController().navigate(action)
```

Εικόνα 139. Τμήμα κώδικα για την πλοήγηση μεταξύ προορισμών με τη χρήση του NavController, μεταφέροντας δεδομένα μεταξύ τους – Στιγμιότυπο οθόνης 2

6.4.1.5. Room²⁹

Η Room αποτελεί βιβλιοθήκη της Jetpack. Παρέχει μία βελτιωμένη λύση στη χρήση της SQLite βάσης δεδομένων, βασιζόμενη σε αυτήν, αλλά με κάποιες διαφοροποιήσεις. Οι διαφοροποιήσεις αυτές συνίστανται στον έλεγχο εγκυρότητας, κατά τη μεταγλώττιση, των *SQL queries* που συντάσσονται, ώστε να εξασφαλίζεται η απουσία σφαλμάτων κατά τη λειτουργία της εφαρμογής. Επίσης, χάρη στη χρήση *annotations*, περιορίζεται η επανάληψη κώδικα και μειώνονται τα σφάλματα. Τα *annotations* είναι εξαιρετικά χρήσιμα, διότι παρέχουν αυτόματη ανανέωση του περιεχομένου λ.χ. των *SQL queries*, σε περίπτωση που πραγματοποιηθεί κάποια αλλαγή στο σχήμα της βάσης. Ένας τρίτος παράγοντας είναι η δυνατότητα *migration*, χάρη στην οποία ο προγραμματιστής καλείται να ενημερώσει για τις αλλαγές που πραγματοποιήθηκαν, και να αποτρέψει σφάλματα και ασυνέπειες που μπορεί να προέκυψαν κατά τις αλλαγές.

Τα κύρια στοιχεία της Room αφορούν στα παρακάτω:

- *Database class*: Αναφέρεται στη βάση και δημιουργεί τη σύνδεση μεταξύ των μόνιμων δεδομένων (*persisted data*) τη εφαρμογής. Συμβολίζεται με `@Database` και περιλαμβάνει τη λίστα των *Entities* και των *DAO*.

Παρακάτω παρατίθεται η υλοποίησή της στην εφαρμογή μας:




```

@Database(
    entities = [AvatarEntity::class, ProductEntity::class, RecycledProductEntity::class, UserEntity::class],
    version = 1
)
abstract class AppDatabase: RoomDatabase() {
    abstract fun userDao(): UserDao
    abstract fun avatarDao(): AvatarDao
    abstract fun productDao(): ProductDao
    abstract fun recycledProductDao(): RecycledProductDao

    companion object {
        private val TAG = AppDatabase::class.java.name

        // the name of the db I'm creating
        private const val DATABASE_NAME = "recycle_it_database"

        @Volatile
        private var INSTANCE: AppDatabase? = null

        private fun buildDatabase(context: Context): AppDatabase {
            Log.i(TAG, msg: "Creating database instance.")
            // applicationContext: so that it is available throughout all the application
            return Room.databaseBuilder(context.applicationContext, AppDatabase::class.java, DATABASE_NAME)
                .allowMainThreadQueries() // allow queries on ui thread
                .build()
        }

        fun getInstance(context: Context): AppDatabase {
            // whatever I do, happens in a sequence (not asynchronously)
            return INSTANCE ?: synchronized(lock: this) {
                INSTANCE ?: buildDatabase(context).also { INSTANCE = it }
            }
        }
    }
}

```

Εικόνα 140. Room - Database Class

Πιο συγκεκριμένα, όπως φαίνεται και στην παραπάνω εικόνα, η τάξη με το annotation `@Database` χρειάζεται να είναι `abstract` και να κληρονομεί από την τάξη `RoomDatabase`. Οι μέθοδοι `userDao()`, `avatarDao()`, `productDao()` και `recycledProductDao()` αποτελούν `abstract` μεθόδους που επιστρέφουν τις αντίστοιχες τάξεις DAO που έχουν οριστεί κατά τη δημιουργία του περιεχομένου της βάσης (βλ. παρακάτω).

Στο εσωτερικό της προαναφερθείσας τάξης δημιουργήθηκε ένα *companion object*, το περιεχόμενο του οποίου είναι προσβάσιμο από τις υπόλοιπες τάξεις της εφαρμογής (έχει δηλαδή λειτουργία παρόμοια με εκείνη ενός *static* αντικειμένου στη *Java*). Εκεί ορίζεται ένα `instance` της τάξης `AppDatabase`, το οποίο κατά τη χρήση στη μέθοδο `getInstance()` βοηθά στην αναφορά στη βάση δεδομένων. Παράλληλα, αποτρέπει την ασύγχρονη κλήση του αντικειμένου, αποτρέποντας ασυνέπειες στη βάση που μπορεί να προκύψουν κατά την ταυτόχρονη πραγματοποίηση διαφορετικών ενεργειών στη βάση. Τέλος, βάση ορίστηκε να τρέχει στο κύριο νήμα (*Main Thread*).

- *Entities*: Οι *Entities* αντιπροσωπεύουν τους πίνακες στη βάση δεδομένων της εφαρμογής μας. Αυτές είναι τέσσερις στην περίπτωση μας: *Product Entity*, *Recycled Product Entity*, *User Entity* και *Avatar Entity*.

Ακολουθεί παράδειγμα της οντότητας προϊόντος:



```
@IgnoreExtraProperties
@Entity(tableName = "product_table")
data class ProductEntity(
    @PrimaryKey(autoGenerate = false)
    @Exclude var barcodeValue: String = "",
    @ColumnInfo(name = "product_name")
    var productName: String = "",
    @ColumnInfo(name = "image_url")
    var imageURL: String = "",
    @ColumnInfo(name = "product_materials")
    var productMaterials: String = "",
    @ColumnInfo(name = "recyclable")
    var recyclable: Boolean = false,
    @ColumnInfo(name = "recycling_info")
    var recyclingInfo: String = "",
    @ColumnInfo(name = "total_recycled_counter")
    var totalRecycledCounter: Int = 0
)
```

Εικόνα 141. Room - Product Entity

Όπως έγινε γνωστό από την παρουσίαση του σχήματος της βάσης στην ενότητα Firebase Realtime Database, η οντότητα προϊόν περιλαμβάνει τα πεδία κωδικός προϊόντος (*barcodeValue*), που αποτελεί το μοναδικό κλειδί του (primary key), το όνομα προϊόντος (*productName*), σύνδεσμος εικόνας προϊόντος (*imageURL*), υλικά προϊόντος, ανακυκλωσιμότητα ή μη (*recyclable*), πληροφορίες/ Συμβουλές για την ανακύκλωση ή μη του προϊόντος (*recyclingInfo*), μετρητής συνολικής ανακύκλωσης από τους χρήστες (*totalRecycledCounter*).

Το annotation *@IgnoreExtraProperties* προστέθηκε για τη διευκόλυνσή μας κατά την αντιστοίχιση των δεδομένων της *Firebase Realtime Database*, η οποία αποτελεί μία *noSQL* βάση δεδομένων, στη *Room Database*, η οποία είναι μία *relational* βάση δεδομένων. Επομένως, ενώ η τιμή του κωδικού προϊόντος αποτελεί ξεχωριστό κόμβο στην πρώτη, αλλά εμπεριέχεται στην οντότητα της δεύτερης, με αυτόν τον τρόπο διασφαλίζεται η αποφυγή συγκρούσεων.

- *DAO (Data Access Objects)*: Πρόκειται για interfaces τα οποία περιέχουν ένα σύνολο μεθόδων υπεύθυνες για την αλληλεπίδραση με τα δεδομένα της βάσης. Οι βασικές μέθοδοι συγκεντρώθηκαν στο *BaseDao*, όπως φαίνεται παρακάτω και περιλαμβάνουν την εισαγωγή, ενημέρωση και διαγραφή των δεδομένων.



```

interface BaseDao<T> {
    @Delete
    fun delete(value: T): Int

    @Delete
    fun deleteAll(vararg values: T): Int

    @Insert(onConflict = OnConflictStrategy.REPLACE)
    fun insert(value: T): Long

    @Insert(onConflict = OnConflictStrategy.ABORT)
    fun insertAll(vararg values: T): List<Long>

    @Update(onConflict = OnConflictStrategy.IGNORE)
    fun update(value: T)

    @Update(onConflict = OnConflictStrategy.IGNORE)
    fun updateAll(vararg value: T)
}

```

Εικόνα 142. Room - BaseDao

Το BaseDao κληρονομείται από το σύνολο των DAOs που δημιουργήθηκαν στην εφαρμογή, προκειμένου να έχουν πρόσβαση στις βασικές λειτουργίες που ορίστηκαν εκεί, ενώ προστίθενται και νέες. Τα DAOs που δημιουργήθηκαν είναι τα εξής: *Avatar Dao*, *Product Dao*, *Recycled Product Dao* και *User Dao*.

Παρακάτω παρατίθεται το *Recycled Product Dao*:

```

@Dao
interface RecycledProductDao: BaseDao<RecycledProductEntity> {
    @Query("SELECT * FROM user_recycled_products_table WHERE uid = :uid")
    fun getAll(uid: String): List<RecycledProductEntity>

    @Query("SELECT COUNT(*) FROM user_recycled_products_table WHERE uid = :uid AND barcode_value = :barcode")
    fun getTimesRecycledByBarcode(uid: String, barcode: String): Int?

    @Query(
        "SELECT product_table.product_name AS productName, product_table.barcodeValue AS barcodeValue, product_table.image_url AS imageURL, " +
        "product_table.recyclable AS isRecyclable, COUNT (product_table.barcodeValue) AS timesRecycled " +
        "FROM product_table INNER JOIN user_recycled_products_table " +
        "ON product_table.barcodeValue = user_recycled_products_table.barcode_value AND user_recycled_products_table.uid = :uid " +
        "AND user_recycled_products_table.recycled_timestamp BETWEEN :timestampStampStart AND :timestampEnd " +
        "GROUP BY product_table.barcodeValue"
    )
    fun getRecycledProductsByDate(uid: String, timestampStampStart: Long, timestampEnd: Long): List<RecycledProduct>

    @Query("DELETE FROM user_recycled_products_table WHERE uid = :uid")
    fun deleteAllForUser(uid: String)
}

```

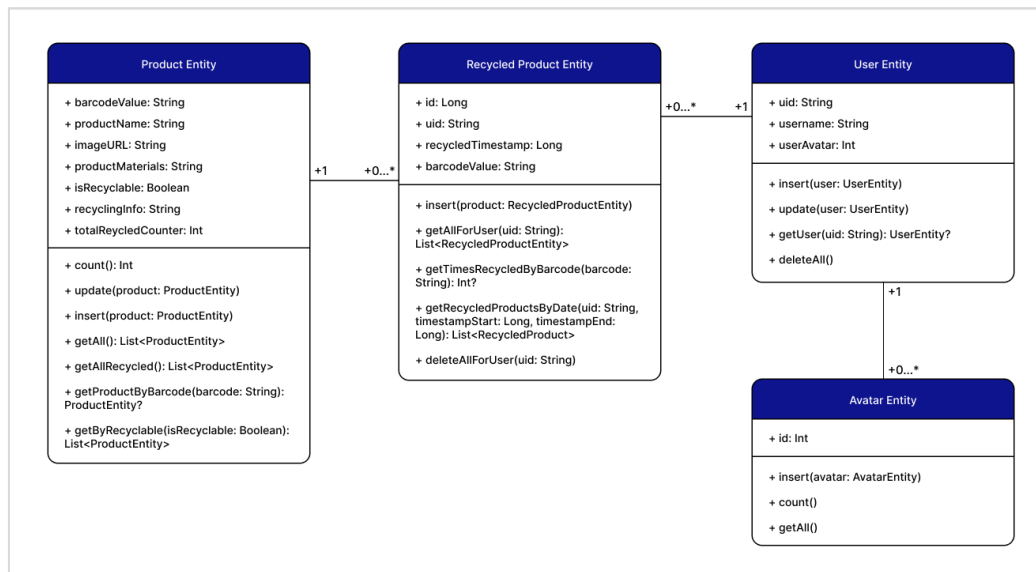
Εικόνα 143. Room - Recycled Product Dao



Τα *queries* που προβάλλονται στην εικόνα παρουσιάζονται αναλυτικά:

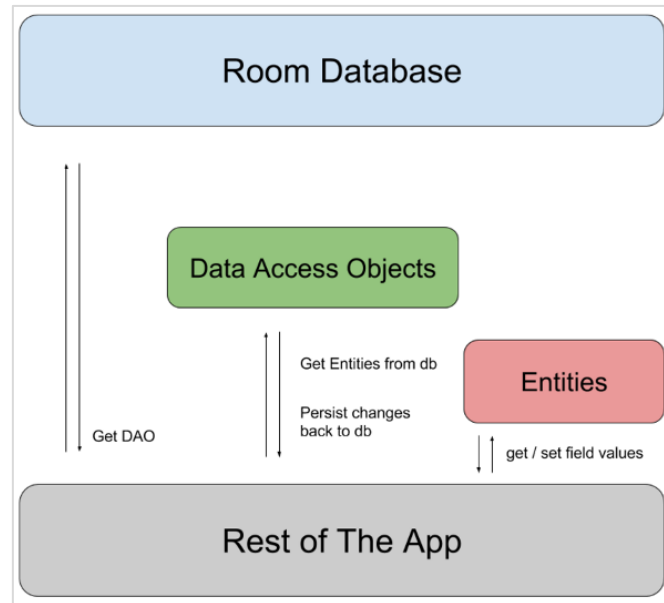
- *getAll()*: Επιστρέφει τη λίστα των προϊόντων που έχουν ανακυκλωθεί από τον δεδομένο χρήστη.
- *getTimesRecycledByBarcode()*: Αφορά στην καταμέτρηση των συνολικών ανακυκλώσεων ενός χρήστη για ένα δεδομένο προϊόν.
- *getRecycledProductsByDate()*: Επιστρέφει τη λίστα των προϊόντων που έχουν ανακυκλωθεί από τον δεδομένο χρήστη, κατά την καθορισμένη ημερομηνία.
- *deleteAllForUser()*: Με τη μέθοδο αυτή διαγράφεται το σύνολο των ανακυκλωμένων προϊόντων του δεδομένου χρήστη.

Ακολουθεί ένα UML διάγραμμα τάξης για την καλύτερη κατανόηση των συστατικών της βάσης:



Εικόνα 144. Διάγραμμα Τάξης UML.

Στην εικόνα που φαίνεται παρακάτω προβάλλεται η αρχιτεκτονική της Room, όπως παρουσιάζεται στην επίσημη ιστοσελίδα για τους [Android developers](#).



Εικόνα 145. Αρχιτεκτονική της Room

Λαμβάνοντας υπόψιν όλα όσα προαναφέρθηκαν, η βιβλιοθήκη της Room χρησιμοποιήθηκε στην εφαρμογή σε συνδυασμό με εκείνη της Firebase, προκειμένου να διασφαλισθεί η μείωση στην χρήση των δεδομένων, τα οποία θα αυξάνονταν με την κάθε κλήση στη Firebase. Ενώ σημαντικό παράγοντα αποτέλεσε και η διευκόλυνση κατά τη δημιουργία των *SQL queries* και η ευκολία στη χρήση τους.

6.4.1.6. CameraX³⁰

Πρόκειται για μία βιβλιοθήκη της Jetpack, η οποία αναφέρεται πως υποστηρίζει άνω του 98% των Android συσκευών, όντας συμβατή αυτές οι οποίες έχουν Android 5.0 (API level 21) και άνω. Παράλληλα, προβάλλεται η συνέπεια στη συμπεριφορά της μεταξύ των συσκευών που αναφέρθηκαν, γεγονός που καθιστά τη δουλειά ενός προγραμματιστή αισθητά ευκολότερη. Βασικά στοιχεία τα οποία χρειάζεται κάποιος να λάβει υπόψιν του καθώς υλοποιεί τη χρήση της κάμερας, όπως ο λόγος διαστάσεων (*aspect ratio*), ο προσανατολισμός της οθόνης (*orientation*), οι διαστάσεις της προεπισκόπησης (*preview size*) και της εικόνας (*image size*) είναι εύκολα υλοποιήσιμα με τη χρήση της CameraX. Ο ισχυρισμός αυτός υποστηρίζεται και από ερευνητικά δεδομένα που προέκυψαν κατά τον έλεγχο της παρούσας βιβλιοθήκης σε πλήθος συσκευών.

Ένα ακόμα σημαντικό πλεονέκτημα είναι η ύπαρξη περιπτώσεων χρήσης (*use cases*) που περιλαμβάνει η CameraX για διαφορετικές λειτουργίες, όπως η Προεπισκόπηση (*Preview*), Ανάλυση Εικόνας (*Image Analysis*), η Λήψη Εικόνας (*Image Capture*) και η Λήψη Βίντεο (*Video Capture*). Καθεμία εκ των οποίων εξυπηρετεί ξεχωριστή λειτουργία, για την υλοποίηση της οποίας δεν χρειάζεται να επιβαρύνεται ο εκάστοτε προγραμματιστής. Στα πλαίσια της παρούσας εφαρμογής έγινε χρήση της Προεπισκόπησης για άμεση ανατροφοδότηση σχετικά με το αντικείμενο που προβάλλεται στην κάμερα, καθώς και η Ανάλυση Εικόνας, η οποία συνδυάστηκε με την υπηρεσία Barcode Scanning του ML Kit, για την ανάγνωση κωδικών barcode σε πραγματικό χρόνο.

6.4.2. Material³¹

Περιέχει μία πληθώρα οδηγιών και κατευθυντήριων γραμμών σχετικά με τις καλύτερες πρακτικές από τη Google, καθώς επίσης και στοιχείων έτοιμων προς χρήση/ υλοποίηση. Με αυτόν τον



τρόπο, διευκολύνεται η σχεδίαση και η υλοποίηση των στοιχείων απαραίτητων για μία εφαρμογή. Τα παραπάνω είναι διαθέσιμα για Android, iOS, Flutter και Web.

Την παρούσα στιγμή, η Material έχει 3 εκδόσεις, εκ των οποίων οι δύο τελευταίες χρησιμοποιήθηκαν στην παρούσα εφαρμογή. Ορισμένα παραδείγματα αποτελούν τα κουμπιά (*Button*³² και *Floating Action Button*³³), τα πεδία συμπλήρωσης (*Text Input Layout*³⁴), ο δείκτης επίδειξης προόδου (*Circular Progress Indicator*³⁵), το μενού πλοήγησης (*Bottom Navigation*³⁶), τα αναδυόμενα παράθυρα (*Dialog*³⁷), και τέλος, τα αναδυόμενα παράθυρα επιλογής ημερομηνίας (*Date Picker*³⁸).

Η Material διαθέτει πλούσιο υλικό στο Figma σχετικά με τα στοιχεία που παρέχονται για τη σχεδίαση μίας εφαρμογής³⁹, όλα έτοιμα προς χρήση. Μία πληθώρα αυτών των υλικών χρησιμοποιήθηκαν για τη σχεδίαση αλλά και την υλοποίηση της παρούσας εφαρμογής. Παραδείγματα φαίνονται στην ενότητα *Εγχειρίδιο Χρήστη*.

6.4.3. Google Fonts⁴⁰

Το επίσημο site της Google Fonts αναφέρει ότι αποτελεί μία βιβλιοθήκη ύψους 1464 γραμματοσειρών ανοιχτού κώδικα, εκ των οποίων χρησιμοποιήθηκε η Roboto⁴¹ για τις ανάγκες της εφαρμογής. Πέραν των γραμματοσειρών, υποστηρίζεται μία πληθώρα εικονιδίων⁴², που αποβήκαν εξαιρετικά χρήσιμα για την υλοποίηση της εφαρμογής.



7. Συμπεράσματα και Μελλοντικές Επεκτάσεις

7.1. Συμπεράσματα

Κατά την ανάπτυξη της εφαρμογής διαπιστώθηκαν ορισμένοι περιορισμοί. Πρώτος είναι ο περιορισμός του προσανατολισμού της οθόνης. Καθώς ο όγκος της πληροφορίας σε ορισμένες από τις οθόνες είναι μεγάλος, με αποτέλεσμα να μην εμφανίζεται εξίσου ικανοποιητικά και στο δύο πιθανούς προσανατολισμούς της οθόνης (portrait mode, landscape mode), αποφασίστηκε πως θα υιοθετηθεί προσέγγιση μόνο για τον έναν εκ των δύο, δηλαδή τον portrait.

Δεύτερο περιορισμό αποτέλεσε η αδυναμία εύρεσης κάποιου API που να περιέχει μία λίστα προϊόντων, με την πληροφορία της ανακυκλωσιμότητάς τους ή μη και επιπρόσθετα την πληροφορία του κωδικού barcode που τους αντιστοιχεί. Έτσι, η βάση δεδομένων που χρησιμοποιείται είναι μικρής εμβέλειας, διότι δημιουργήθηκε από εμάς.

7.2. Μελλοντικές Επεκτάσεις

Κατά την ανάπτυξη της εφαρμογής, διαπιστώθηκαν σημεία βελτίωσης της παρούσας εφαρμογής. Μερικά από αυτά παρατίθενται παρακάτω:

- Θα ήταν χρήσιμο, αν οι ίδιοι οι χρήστες μπορούσαν να καταχωρήσουν ένα προϊόν βάσει του κωδικού barcode που αντιστοιχεί σε αυτό, παρέχοντας παράλληλα πληροφορίες σχετικά με αυτό και τη διαδικασία ανακύκλωσής του. Για την καλύτερη εμπειρία ενδεχομένως θα ήταν προτιμότερο η κάθε καταχώρηση να τίθεται υπό τον έλεγχο κάποιου διαχειριστή ώστε να επιβεβαιώνεται η εγκυρότητά της.
- Ενδιαφέρουσα προοπτική παρουσιάζει η δυνατότητα πώλησης των προϊόντων προς ανακύκλωση (ή και μη), π.χ. βιβλία, εφημερίδες, ρούχα, σε ενδεχόμενους ενδιαφερόμενους. Με αυτόν τον τρόπο θα μπορούσε να αυξηθεί το εύρος των απόμων που εξασκούν την ανακύκλωση, παρέχοντας, δηλαδή, εξωτερικά κίνητρα.
- Η δωρεά ανακυκλώσιμων προϊόντων (ή και μη) είναι μία ακόμα καλή ιδέα, στο ύφος της προηγούμενης, αν και πιο αλτρουιστική, που θα βοηθούσε στην ευκολότερη επαναχρησιμοποίηση προϊόντων. Την πρόταση αυτή θα καθιστούσε ιδανική μία συνεργασία με μη κερδοσκοπικές οργανώσεις, οι ανάγκες των οποίων θα γίνονταν γνωστές μέσω της εφαρμογής.
- Μία εναλλακτική στην πώληση προϊόντων αφορά στη συνεργασία με καταστήματα, όπως για παράδειγμα σούπερμαρκετ, τα οποία θα παρέχουν εκπτώσεις σε προϊόντα τους, αν οι χρήστες παρουσιάζουν πρόοδο στην ανακύκλωσή τους.
- Ένα σημείο που θα διευκόλυνε την ανακύκλωση, θα ήταν η προσθήκη πληροφοριών σχετικά με την περιοχή στην οποία θα μπορούσε ο χρήστης να ανακυκλώσει το εκάστοτε προϊόν.
- Τέλος, η δυνατότητα αναφοράς των προβλημάτων που αντιμετωπίζουν κατά την ανακύκλωση στην περιοχή τους, φαίνεται εξαιρετικής σημασίας.



8. Βιβλιογραφία

- ¹ Διαθέσιμη στον διαδικτυακό ιστότοπο: <https://www.isri.org/news-publications/news-details/2018/10/10/poll-if-americans-are-expected-to-recycle-it-needs-to-be-made-easy>
- ² Handbook of Recycling: State-of-the-art for Practitioners, Analysts, and Scientists – Ernst Worrell and Markus A. Reuter. Διαθέσιμο στον σύνδεσμο: <https://www.sciencedirect.com/book/9780123964595/handbook-of-recycling>
- ³ <https://www.rts.com/blog/recycling-facts-statistics/>
- ⁴ <https://www.brysonrecycling.org/recycling/why-recycle/>,
<https://libre.stanford.edu/pssistanford-recycling/frequently-asked-questions/frequently-asked-questions-benefits-recycling>,
https://www.ecocycle.org/files/pdfs/why_recycle_%20brochure.pdf,
<http://www.slwp.org.uk/what-we-do/recycling-composting/why-it-is-important-to-recycle-and-compost/>, <https://www.sappi.com/faqs-recycling>, <https://www.herrco.gr/polites/benefits/>
- ⁵ Διαθέσιμη στον ιστότοπο: <https://followgreen.gr/>
- ⁶ Διαθέσιμη στο Google PlayStore.
- ⁷ Διαθέσιμη στον ιστότοπο: <https://www.herrcoglass.gr/> και στο Google PlayStore.
- ⁸ Διαθέσιμο στον ιστότοπο: <https://recyclecoach.com/residents/recyclepedia/>, και στο Google PlayStore. Οι εικόνες προέρχονται από την παρουσίαση της εφαρμογής στο Google PlayStore.
- ⁹ Διαθέσιμο στον ιστότοπο: <https://www.recyclesmart.com/> και στο Google PlayStore.
- ¹⁰ Οι εικόνες που παρατίθενται προέρχονται από την παρουσίαση της εφαρμογής στο Google PlayStore.
- ¹¹ Διαθέσιμο στον ιστότοπο: <https://trashly.io/>.
- ¹² Η προέλευση της εικόνας που προβάλλεται στην οθόνη είναι <https://dribbble.com/shots/10742148/attachments/2410128?mode=media>
- ¹³ Effective Kotlin, *Best Practices* – Marcin Moskala, Published by: Leanpub, on 2020
- ¹⁴ <https://kotlinlang.org/>
- ¹⁵ <https://developer.android.com/studio/intro>
- ¹⁶ <https://firebase.google.com/products-build>
- ¹⁷ <https://firebase.google.com/docs/crashlytics>
- ¹⁸ <https://firebase.google.com/docs/auth>
- ¹⁹ <https://firebase.google.com/docs/database>
- ²⁰ <https://developers.google.com/ml-kit/vision/barcode-scanning>
- ²¹ Διαθέσιμο στον σύνδεσμο: <https://firebase.google.com/docs/ml-kit/read-barcodes>
- ²² Διαθέσιμο στον σύνδεσμο: <https://www.figma.com/>
- ²³ Διαθέσιμο στον σύνδεσμο: <https://lottiefiles.com/>
- ²⁴ Διαθέσιμο στον σύνδεσμο: <https://developer.android.com/jetpack>
- ²⁵ Διαθέσιμο στον σύνδεσμο: <https://developer.android.com/topic/libraries/view-binding#kts>
- ²⁶ Διαθέσιμο στους συνδέσμους: <https://developer.android.com/jetpack/androidx/releases/fragment>,
<https://developer.android.com/guide/fragments>
- ²⁷ Διαθέσιμο στον σύνδεσμο: <https://developer.android.com/guide/fragments/dialogs>
- ²⁸ Διαθέσιμο στον σύνδεσμο: <https://developer.android.com/guide/navigation>
- ²⁹ Διαθέσιμο στον σύνδεσμο: <https://developer.android.com/training/data-storage/room>
- ³⁰ Διαθέσιμο στον σύνδεσμο: <https://developer.android.com/training/camerax>
- ³¹ Διαθέσιμο στους συνδέσμους: <https://m2.material.io/>, <https://m3.material.io/>
- ³² Διαθέσιμο στον σύνδεσμο: <https://m3.material.io/components/buttons/overview>



-
- ³³ Διαθέσιμο στον σύνδεσμο: <https://m3.material.io/components/floating-action-button/overview>
- ³⁴ Διαθέσιμο στον σύνδεσμο: <https://m3.material.io/components/text-fields/overview>
- ³⁵ Διαθέσιμο στον σύνδεσμο: <https://m2.material.io/components/progress-indicators>
- ³⁶ Διαθέσιμο στον σύνδεσμο: <https://m2.material.io/components/bottom-navigation#usage>
- ³⁷ Διαθέσιμο στον σύνδεσμο: <https://m3.material.io/components/dialogs/overview>
- ³⁸ Διαθέσιμο στον σύνδεσμο: <https://m2.material.io/components/date-pickers>
- ³⁹ Διαθέσιμο στον σύνδεσμο: <https://www.figma.com/community/file/778763161265841481>,
<https://www.figma.com/community/file/1035203688168086460>,
<https://www.figma.com/community/file/1001534940974132136>
- ⁴⁰ Διαθέσιμο στον σύνδεσμο: <https://fonts.google.com/?icon.platform=android>
- ⁴¹ Διαθέσιμο στον σύνδεσμο: <https://fonts.google.com/specimen/Roboto>
- ⁴² Διαθέσιμο στον σύνδεσμο: <https://fonts.google.com/icons>

