



ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ

ΤΜΗΜΑ ΔΙΟΙΚΗΤΙΚΗΣ ΕΠΙΣΤΗΜΗΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ

Ανάπτυξη και Σχεδίαση Κινητών Εφαρμογών:

TrackIt

ΜΕΛΗ ΟΜΑΔΑΣ:		
ΕΥΑΓΓΕΛΙΑ ΚΑΠΕΤΑΝΑΚΗ	8210050	t8210050@aueb.gr
ΑΓΓΕΛΟΣ ΙΩΑΝΝΗΣ ΚΗΡΥΚΟΣ	8210057	t8210057@aueb.gr
ΓΡΗΓΟΡΙΟΣ ΣΥΜΕΩΝ ΣΑΣΣΑΝΗΣ	8210222	t8210222@aueb.gr
ΑΓΓΕΛΙΚΗ ΤΣΑΓΚΑΡΑΚΗ	8210150	t8210150@aueb.gr

Καθηγητής: Μάριος Φραγκούλης

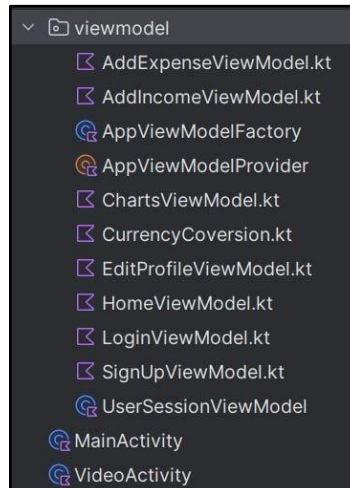
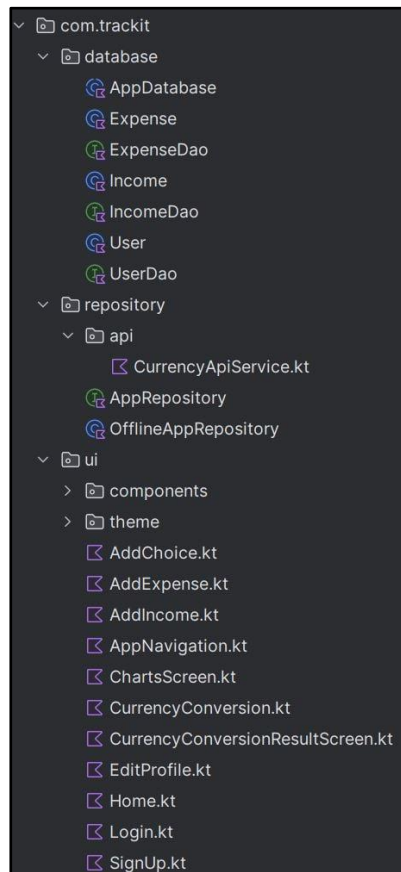
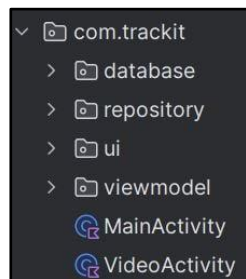
Αθήνα 2025

Τεχνολογίες που χρησιμοποιήθηκαν:

Η εφαρμογή TrackIt μέσω του Android Studio βασίζεται στην γλώσσα προγραμματισμού **Kotlin**, στο build tool **Gradle** και το framework **Jetpack Compose** για την κατασκευή της διεπαφής χρήστη. Υιοθετεί σύγχρονες αρχιτεκτονικές πρακτικές, κάνοντας χρήση σημαντικών βιβλιοθηκών όπως το **Android Room**, το **Navigation Component** και το **MPAndroidChart** για τη διαχείριση βάσεων δεδομένων, πλοήγηση, και οπτικοποίηση δεδομένων αντίστοιχα. Παρακάτω ακολουθεί μια ανάλυση της αρχιτεκτονικής και της δομής του έργου.

Αρχιτεκτονική Εφαρμογής

Η εφαρμογή υλοποιεί το αρχιτεκτονικό πρότυπο **MVVM (Model-View-ViewModel)** και περιλαμβάνει σαφή διαχωρισμό ευθυνών στα διάφορα επίπεδα, όπως φαίνεται παρακάτω:



1. Βάση Δεδομένων (Database Layer)

Ο φάκελος database περιέχει την υλοποίηση της τοπικής βάσης δεδομένων χρησιμοποιώντας το **Android Room**, που βασίζεται στη SQLite:

- **AppDatabase.kt**: Η κύρια κλάση που ορίζει τη βάση δεδομένων και κάνει το binding με τα DAOs.
- **Expense.kt, Income.kt, User.kt**: Τα **Entities** που αντιπροσωπεύουν τους πίνακες της βάσης δεδομένων.
- **ExpenseDao.kt, IncomeDao.kt, UserDao.kt**: Τα **Data Access Objects (DAOs)** που διαχειρίζονται τις CRUD (create , read, update , delete) λειτουργίες για κάθε entity. Είναι υπεύθυνα για την επικοινωνία με τη βάση δεδομένων.

Η χρήση του Room διασφαλίζει ασφαλή και ευέλικτη διαχείριση της SQLite, ενώ τα DAOs προσφέρουν σαφή και επεκτάσιμο τρόπο αλληλεπίδρασης με τη βάση δεδομένων.

2. Repository Layer

Ο φάκελος repository περιέχει τον κώδικα για την επικοινωνία με τη βάση δεδομένων και το δίκτυο:

- **CurrencyApiService.kt**: Υλοποιεί την επικοινωνία με την API για την μετατροπή νομισμάτων.
- **OfflineAppRepository.kt**: Είναι υπεύθυνο για την ενδιάμεση λογική, συνδυάζοντας δεδομένα από την τοπική βάση και εξωτερικές πηγές. Υποστηρίζει **offline mode**, διασφαλίζοντας τη συνέχιση λειτουργίας ακόμη και χωρίς σύνδεση στο διαδίκτυο.

Το repository λειτουργεί ως ενδιάμεσος ανάμεσα στο ViewModel και στα δεδομένα, διαχωρίζοντας τη λογική της εφαρμογής από την αποθήκευση ή ανάκτηση των δεδομένων.

3. ViewModel Layer

Στον φάκελο viewmodel, φιλοξενούνται όλες οι κλάσεις ViewModel, οι οποίες είναι υπεύθυνες για την κατάσταση της εφαρμογής:

- **AddExpenseViewModel, AddIncomeViewModel**: Διαχειρίζονται τα δεδομένα και τη λογική για την προσθήκη εξόδων/εσόδων.
- **HomeViewModel, ChartsViewModel**: Υποστηρίζουν την αρχική σελίδα και την οπτικοποίηση δεδομένων μέσω γραφημάτων.
- **CurrencyConversionViewModel**: Αντιπροσωπεύει τη λογική για τη μετατροπή νομισμάτων.

- **EditProfileViewModel, LoginViewModel, SignUpViewModel:** Διαχειρίζονται τις λειτουργίες χρήστη, όπως σύνδεση, εγγραφή και επεξεργασία προφίλ.
- **UserSessionViewModel:** Παρέχει στις σελίδες το user id του συνδεδεμένου χρήστη
- **AppViewModelFactory, AppViewModelProvider:** Διευκολύνουν τη δημιουργία ViewModels έτσι ώστε να μπορούν να δέχονται ως παράμετρο το Id του συνδεδεμένου χρήστη και να συνδεθούν με το Repository

Το ViewModel Layer χρησιμοποιεί **StateFlow** για τη διαχείριση των δεδομένων και την ενημέρωση της διεπαφής χρήστη σε πραγματικό χρόνο.

4. Διεπαφή Χρήστη (UI Layer)

Ο φάκελος ui περιέχει την υλοποίηση των οθονών και των επαναχρησιμοποιούμενων στοιχείων διεπαφής χρήστη, κατασκευασμένων με **Jetpack Compose**:

- **Οθόνες:**
 - **AddChoice.kt, AddExpense.kt, AddIncome.kt:** Επιτρέπουν την εισαγωγή δεδομένων εσόδων και εξόδων.
 - **Home.kt:** Παρουσιάζει συνοπτικά δεδομένα (με χρήση λιστών).
 - **ChartsScreen.kt:** Παρουσιάζει γραφήματα και στατιστικά δεδομένα με χρήση του **MPAndroidChart** (με χρήση λιστών).
 - **CurrencyConversion.kt, CurrencyConversionResultScreen.kt:** Υποστηρίζουν τη μετατροπή νομισμάτων και την προβολή αποτελεσμάτων.
 - **Login.kt, SignUp.kt:** Διαχειρίζονται τη σύνδεση και την εγγραφή χρηστών.
 - **EditProfile.kt:** Υποστηρίζει τη διαχείριση του προφίλ χρήστη.
- **Components και Theme:**
 - **components/:** Περιέχει επαναχρησιμοποιούμενα UI components (π.χ., κουμπιά, φόρμες).
 - **theme/:** Ορίζει τα θέματα, τα χρώματα και τα στυλ της εφαρμογής.

Η χρήση του **Jetpack Compose** διευκολύνει την κατασκευή δυναμικών UI με λιγότερο boilerplate κώδικα.

5. Navigation

Το **AppNavigation.kt** υλοποιεί την πλοήγηση της εφαρμογής χρησιμοποιώντας το **Navigation Component**, εξασφαλίζοντας ομαλή μετάβαση μεταξύ των οθονών.

6. Activity

MainActivity.kt: Αρχικοποιεί την εφαρμογή όταν την πατήσει ο χρήστης στο κινητό του.

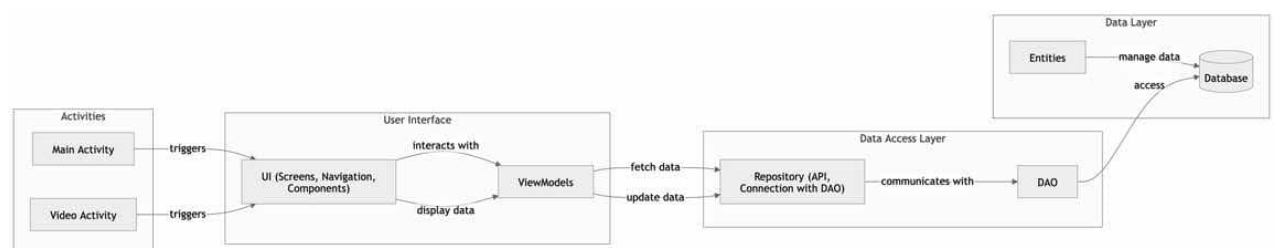
VideoActivity.kt: Αναπαράγει ένα τοπικό βίντεο (που είναι αποθηκευμένο στους πόρους της εφαρμογής). Χειρίζεται σφάλματα και κλείνει την οθόνη μόλις ολοκληρωθεί ή παρουσιαστεί κάποιο πρόβλημα κατά την αναπαραγωγή.

7. Οπτικοποίηση Δεδομένων

Η χρήση της βιβλιοθήκης **MPAndroidChart** στον φάκελο ui (μέσω του ChartsScreen.kt) παρέχει γραφήματα και στατιστικά για τα δεδομένα του χρήστη. Πιο συγκεκριμένα, περιλαμβάνει γραφήματα ράβδων για την ανάλυση των εξόδων.

8. UML Diagram

Παρακάτω παραθέτουμε και ένα **UML διάγραμμα** για τη σύνδεση των στοιχείων της εφαρμογής:



9. Authors

Συμβολή κάθε μέλους στην υλοποίηση της εφαρμογής:

Name: Ευαγγελία Καπετανάκη

Role: Author

Contributions:

- Navigation
- Pages' debugging

Screens Developed:

- Home.kt
- Expenses.kt
- AppNavigation.kt
- AddExpense.kt
- AddChoice.kt
- ExpensesDao.kt
- HomeViewModel.kt
- AddExpensesViewModel.kt
- MainActivity.kt
- AppRepository.kt
- OfflineAppRepository.kt

Name: Άγγελος-Ιωάννης Κηρύκος

Role: Author

Contributions:

- Σύνδεση εφαρμογής με τη βάση δεδομένων

Screens Developed:

- AppDatabase
- User
- UserDao
- Income
- IncomeDao
- Login.kt
- SignUp.kt
- AddIncome.kt
- AddIncomeViewModel.kt
- LoginViewModel.kt
- SignUpViewModel.kt
- VideoActivity.kt

Name: Γρηγόριος-Συμεών Σασσάνης

Role: Author

Contributions:

- Υλοποίηση δυναμικών γραφημάτων

Screens Developed:

- EditProfileViewModel.kt
- ChartsViewModel.kt
- ChartsScreen.kt
- EditProfile.kt
- AppViewModelFactory.kt , AppViewModelProvider.kt
- UserSessionViewModel.kt

Name: Αγγελική Τσαγκάρακη

Role: Author

Contributions:

- Σύνδεση με API(api.exchangeratesapi.io) για μετατροπή νομισμάτων και προβολή αποτελεσμάτων

Screens Developed:

- CurrencyConversion.kt
- CurrencyConversionResultScreen.kt
- CurrencyConversionViewModel.kt
- CurrencyApiService.kt
- AddExpense.kt
- AppViewModelFactory.kt
- OfflineAppRepository.kt

Challenges Addressed:

- Δυσκολία στη λειτουργία του **Android Studio**, που δεν μπορούσε να πραγματοποιήσει **build** λόγω προβλημάτων με τη **Gradle**.

Αρχικά, προσπαθήσαμε να εντοπίσουμε το ζήτημα με ελέγχους στις ρυθμίσεις του **Gradle** και του έργου, αλλά το πρόβλημα παρέμενε. Μετά από

διερεύνηση, διαπιστώθηκε ότι το πρόβλημα σχετιζόταν με την **έκδοση της Java**, η οποία δεν ήταν συμβατή με την απαιτούμενη έκδοση για τη Gradle του συγκεκριμένου project.

Η λύση ήρθε με την **αναβάθμιση της Java** στην κατάλληλη έκδοση, που ήταν συμβατή με το build tool της Gradle.

10. Προκλήσεις

1) Χρήση API για μετατροπή νομισμάτων:

Το API που επιλέξαμε, δηλαδή το `api.exchangeratesapi.io`, υποστήριζε αποκλειστικά τη μετατροπή από ευρώ προς άλλα νομίσματα. Για να υποστηρίξουμε όλες τις πιθανές μετατροπές, απαιτήθηκε η λήψη του συναλλαγματικού δείκτη (exchange rate) μέσω του API και στη συνέχεια η ανάπτυξη πρόσθετων υπολογισμών στον κώδικα, ώστε να προκύπτουν τα σωστά αποτελέσματα.

2) Υλοποίηση γραφημάτων με MPAndroidChart:

Η δημιουργία διαδραστικών γραφημάτων για την απεικόνιση δεδομένων υπήρξε απαιτητική, λόγω της πολυπλοκότητας του εργαλείου και της ανάγκης προσαρμογής στις απαιτήσεις της εφαρμογής. Απαιτήθηκε σημαντική εξοικείωση με τη βιβλιοθήκη, καθώς και λεπτομερής παραμετροποίηση για την επίτευξη του επιθυμητού αποτελέσματος.

11. Καινοτομίες

Το **TrackIt** ξεχωρίζει χάρη στην ενσωμάτωση του **MPAndroidChart**, που προσφέρει μια καινοτόμο προσέγγιση στην παρακολούθηση οικονομικών δεδομένων. Μέσα από διαδραστικά και οπτικά ελκυστικά γραφήματα, οι χρήστες μπορούν να αναλύσουν τις δαπάνες και τις τάσεις τους με τρόπο που κάνει την οικονομική διαχείριση πιο κατανοητή και προσιτή. Η εφαρμογή συνδυάζει τη λειτουργικότητα με την ευκολία χρήσης, δίνοντας στους χρήστες ένα προηγμένο εργαλείο για να λαμβάνουν τεκμηριωμένες αποφάσεις σχετικά με τα οικονομικά τους.