



מכון טכנולוגי חולון
Holon Institute of Technology

פיתוח אפליקציות אנדרואיד בשפת קוטלין

Developing Android Applications with Kotlin

מדעים	שם הפקולטה:
מדעי המחשב	המחלקה:
מר ערן קצב	שם המרצה:
הרצאה + תרגול	מתכונת הקורס:
הרצאה – 3 שעות + תרגול 1 שעה	אופן הוראה:
תואר ראשון	שנת לימודים:
3.5 נ"ז	נקודות זכות:
תכנות מונחה עצמים 61307	דרישות קדם:

א. מטרת הקורס:

- היכרות מעמיקה עם אנדרואיד, מערכת ההפעלה המובילה בעולם.
- למידת שפת התכנות Kotlin תוך שימת דגש על תכנות פונקציונלי מסדר גבוה ותכנות מונחה עצמים.
- היכרות מעשית עם ה Coroutines בשפת Kotlin המאפשרים עבודה מקבילית.
- הכרת כלל אבני הבניין הן הוויזואליים והם הלוגים המשמשים לבניית אפליקציית אנדרואיד.
- הקניית היכולת לעבוד עם מסדי נתונים מקומיים ומרוחקים.
- מתן היכולת לתקשורת מלאה עם רכיבי וחיישני המכשיר הנייד השונים.
- מתן היכולת לתקשורת מלאה עם מערכת ההפעלה לתזמון משימות ושימוש בשרותיה השונים.
- תכנון מבני ויסודי של חלקי האפליקציה השונים תוך יישום תבניות עיצוב שנלמדו בקורס ובעבר.
- הקניית יכולת מעשית אשר תבוא לידי ביטוי ביכולת ליצור פרויקטים מורכבים באופן עצמאי.

ב. הרציונל:

מזה חמש שנים שאנדרואיד, פרויקט קוד הפתוח מבית גוגל, הפך למערכת ההפעלה המובילה בעולם. מערכת הפעלה זו היא השער לכמעט כל המכשירים החכמים המקיפים אותנו ומלווים אותנו בכל פעולותינו היומיומיות.

יחד עם זאת הכשרה מוסדית ויסודית כזו העושה שימוש בסביבת הפיתוח המקורית של אנדרואיד, כמעט ואינה קיימת והדבר מתבטא בקושי הקיים בשוק העבודה באיתור מפתחים בעלי ראייה כוללת של המערכת. כמו כן, עם עליית מורכבותה של הסביבה החיצונית עלתה גם מורכבות וחשיבות ארכיטקטורת האפליקציה, וחשיבות תכנונה מראש על חלקיה השונים גבר על העבודה נמוכה והמורכבת עם ממשק המשתמש, שכיום מגיע ברכיבים מובנים ומוכנים מראש לשימוש המתכנתים.

מטרת הקורס היא להקנות לסטודנטים ידע והבנה מעמיקה של שפת קוטלין התמציתית והמותאמת למערכות זמן אמת, תוך יישומה על הפלטפורמה המובילה בעולם במטרה לתת כלי פרקטי למען השתלבות מיטבית בשוק העבודה. הקורס יעניק לסטודנטים את היכולת לפתח אפליקציות היכולות לעשות שימוש כמעט בלתי מוגבל ביכולות במכשיר הנייד, הן ברמה הנמוכה ביותר מבחינת תקשורת עם חומרת המכשיר ומערכת ההפעלה והן הגבוהה ביותר מבחינת שימוש בתבניות עיצוב מתקדמות לפי הסטנדרטים הגבוהים ביותר הנהוגים כיום בתעשייה ומוכתבים על ידי גוגל העולמית.

ג. תיאור תמציתי של הקורס ושיטת ההוראה

הקורס הינו קורס מעשי בו נלמד תוך כדי יישום את שפת הפיתוח לאנדרואיד ושלל רכיבי מערכת ההפעלה העומדים לרשות המפתחים, כך שהתאוריה תשולב בתוך העשייה, כל רעיון יוצג יחד עם יישומו בפועל - בקוד. במהלך הקורס כל סטודנט יצטרך להגיש תרגילים שהינם אפליקציות בסיסיות הכוללות חקר ולמידה עצמית של בנייה ועיצוב ממשק המשתמש וקבלת האירועים השונים ממנו, ופרויקט גמר אשר נעשה בזוגות או בשלשות ויהווה אפליקציית אנדרואיד ברמת השוק. חשוב להדגיש כי בגלל שנושא ממשק המשתמש הינו פשוט יחסית אך חשוב, לכל תרגיל יתווסף מדריך מלא ללימוד עצמי. הפרויקט הסופי יהיה אפליקציה המקיימת אינטראקציה עם מסד נתונים מקומי ומרוחק, אשר נבנתה על פי ארכיטקטורת ה Jetpack וכוללת את מרבית רכיבי האנדרואיד שנלמדו בכיתה.

ד. תכנית ההוראה מפורטת לפי נושאים

תכנית שבועית

שבוע	נושאי הלימוד
1	מבוא למערכת ההפעלה אנדרואיד והיכרות עם שפת קוטלין
2	תכנות פונקציונלי ותכנות מסדר גבוה בשפת קוטלין
3	תכנות מונחה עצמים בשפת קוטלין
4	תכנות אסינכרוני באמצעות Coroutines
5	אבני הבניין לאפליקציית אנדרואיד – חלק 1 Activities, Intents, Permissions, Lifecycle and Persistent storage
6	אבני הבניין לאפליקציית אנדרואיד – חלק 2 Fragments & Adapters
7	MVVM (Model View View-Model) and Live Data
8	מסדי נתונים מקומיים עם ROOM ויצירת פרויקט ה Jetpack הראשון
9	שרותי הטלפון השונים
10	הזרקת תלויות עם Hilt
11	פרויקט ארכיטקטורה מלא הכולל מסדי נתונים מרוחקים באמצעות Retrofit 2
12	מערכת השדרים וההתראות (Broadcast Receivers & Alarm Manager)
13	שירותים ללא ממשק משתמש ותזמון משימות באמצעות המערכת (Services, Job Services) (and Work Manager)

ה. מרכיבי הציון הסופי:

- 30% - תרגילי הגשה במהלך הסמסטר
- 70% - פרויקט גמר (אפליקציה ברמת שוק)
- כדי לעבור את הקורס יש לקבל ציון עובר הן בתרגילי ההגשה והן בפרויקט הגמר.
- פרויקט הגמר של הקורס הוא כ-100 שעות ונעשה בזוגות או בשלשות. מטרת הפרויקט להקנות לסטודנטים ניסיון מעשי בפיתוח אפליקציות וצבירת ניסיון ביציאה לשוק התעסוקה.

ו. ביבליוגרפיה:

- Kotlin in action (Manning) by Dmitry Jemerov and Svetlana Isakova (19 February 2017)
- Mastering Kotlin: Learn advanced Kotlin programming techniques to build apps for Android, iOS, and the web (Packt Publishing) by Nate Ebel (October 11, 2019)
- How to build Android apps with Kotlin (packt publishing) by Alex Forrester (February 26, 2021)
- Kotlin Coroutines by Tutorials (Second Edition): Mastering Coroutines in Kotlin and Android by ray wenderlich Tutorial Team

ז. סילבוס מפורט לפי הנושאים:

Android Jetpack Overview & Intro to Kotlin

- 📌 Introduction – Android Native Development
- 📌 Android OS Historical Overview
- 📌 AndroidX & Android Jetpack Components
- 📌 Android Platform Architecture
- 📌 JVM, DVM and ART
- 📌 Kotlin overview
- 📌 Kotlin's Extensions
- 📌 Kotlin vals & vars
- 📌 Kotlin's Nullability
- 📌 Kotlin Basics (if, when, loops, Arrays, Collections)

Kotlin – Functional Programming

- 📌 Functions & Lambdas
- 📌 Default Arguments
- 📌 High order Functions
- 📌 Collections high order functions (filter, map, flatMap...)
- 📌 Lazy Sequences

- ✚ Scope Functions
- ✚ infix functions
- ✚ extension functions

Object Oriented with Kotlin

- ✚ Classes - Constructors and Properties
- ✚ Inheritance
- ✚ Interfaces and Abstract classes
- ✚ Data classes
- ✚ Delegated properties & lazy initialization
- ✚ Object expressions and Object Declarations
- ✚ SAM interfaces
- ✚ Nested & inner classes
- ✚ Access modifiers and Generics
- ✚ Operator Overloading

Asynchronous programming with Coroutines

- ✚ Intro to Android Views and Bindings
- ✚ Making a Responsive UI
- ✚ Suspension over Blocking
- ✚ CoroutineScope & CoroutineContext
- ✚ Dispatchers
- ✚ Coroutines Builders (launch, async, withContext...)
- ✚ Combining Suspended Functions
- ✚ Jetpack Components Scopes
- ✚ Continuation Object

Android Building Blocks - Part 1

- ✚ Android App Components
- ✚ Activities & Intents (Explicit and Implicit)
- ✚ Runtime Permissions using Launchers

- Activity LifeCycle and Shared preferences

Android Building Blocks - Part 2

- Single Activity Architecture with Fragments
- Fragment Lifecycle
- Using Navigation for Fragments transactions
- Recycler View (Adapter, ViewHolder, LayoutManager)

View Models & Live Data

- App Resources & Localization (language, screen, orientation...)
- Configuration changes & savedInstanceState bundle
- MVVM Design pattern
- Jetpack Lifecycle object
- View Model Scope
- LiveData & MutableLiveData
- Saved state module
- Transformations
- View Model provider Factory

MidProject using Room

- Internal Database with Room
- Activity for Result with Launchers
- MVVM project with ViewModel, LiveData, Navigation and Room.

Phone services

- Google play services & Fused location
- Working with the phone Camera
- Android External Storage
- File Provider & Content Providers

Dependency Injection with Hilt

- The concept of Dependency Injection
- Manual vs Automatic Injection
- Static vs Dynamic injection
- Dagger 2 and Hilt
- Hilt Annotations: @Inject, @Binds and @Provides
- Hilt Modules and @Installin annotation
- Dependencies scope
- View Models with Hilt

Fully Jetpacked Architecture App with local and remote data sources

- Useful external libraries: Glide, Volley, Gson & Retrofit 2
- Repository - for single data entry point to data sources
- Creating Full Jetpack Architecture app using Hilt and Retrofit 2

The Broadcasting system

- Notifications
- Broadcast Receivers
- Static vs Dynamic Registration
- Local Broadcast Manager
- Alarm Manager

Non-UI Related tasks

- Service and Foreground service
- Job Schedulers
- Work Managers
- Background Execution limits