Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Department of computer systems software

MOBILE APPLICATION DEVELOPMENT

Syllabus

Details of the academic discipline

Level of higher education	***First (Bachelor)***
Branch of knowledge	*12 Information technologies*
Specialty	*121 Software engineering*
Educational program	*Software Engineering of Multimedia and Information-Retrieval Systems*
Discipline status	*Selective*
Form of education	*Full-time*
Year of training, semester	*3rd year of training, 5th semester*
Scope of the discipline	*Lectures: 36 hours, computer workshop: 18 hours, independent work: 66 hours.*
Semester control/ control measures	*Credit, modular control work, calendar control*
Lessons schedule	*According to the schedule for the autumn semester of the current academic year (http://roz.kpi.ua/)*
Language of teaching	*English*
Information about the course leader / teachers	Lecturer: *assistant Department of system programming and specialized computer systems (SPiSKS) Radchenko Kostiantyn Oleksandrovych, radchenko.kostiantyn@lll.kpi.ua* Computer works:*Radchenko Kostiantyn Oleksandrovych*
Placement of the course	Google classroom. Access is granted to registered students.

Program of educational discipline

1. Description of the educational discipline, its purpose, subject of study and learning outcomes

Studying the discipline "Mobile Application Development" allows students to develop the competencies necessary for working with modern software, a systematic approach to solving engineering and technical problems using a PC, searching and processing information using modern technologies. Mastering the capabilities and tools of developing applications for mobile applications is a key task of the discipline.

***The purpose** studying the discipline "Mobile Application Development" is the formation of students of a modern level of information and computer culture, knowledge of the structure and principles of building technologies for creating applications based on modern mobile platforms, used in computer systems, and being able to draw correct conclusions from the information received.*

***Subject** the discipline "Mobile Application Development" is information technologies for ensuring the processes of development and improvement of mobile software applications.*

Studying the discipline "Mobile Application Development" strengthens the professional competences (PC) in students, necessary for solving practical tasks of professional activity:
***PC01** Ability to identify, classify and formulate software requirements.*
***PC08** Ability to apply fundamental and interdisciplinary knowledge to successfully solve software engineering problems.*
***PC13** Ability to reasonably select and master software development and maintenance tools.*

PC17 *Ability to develop software for information retrieval systems.*
PC19 *Ability to develop software for multimedia and mulsemedia systems.*

Studying the discipline "Mobile Application Development" contributes to the formation of program learning outcomes (PLO) for students according to the educational program:
PLO01 *To analyze, purposefully search and select the necessary information and reference resources and knowledge to solve professional problems, taking into account modern advances in science and technology.*
PLO08 *To know and to be able to develop a human-machine interface.*
PLO12 *To apply effective approaches to software design in practice.*
PLO15 *To choose programming languages and development technologies to solve the problems of creating and maintaining software.*
PLO18 *To know and be able to apply information technology of processing, storage and transmission of data.*
PLO19 *To know and be able to apply software verification and validation methods.*
PLO38 *To be able to apply programming technologies for multimedia and information retrieval systems software development.*

2. Pre-requisites and post-requisites of the discipline (place in the structural and logical scheme of training according to the relevant educational program)

The successful study of the discipline "Mobile Application Development" is preceded by the study of the disciplines "Mathematical support of multimedia and information retrieval systems", "Algorithmic support of multimedia and information retrieval systems", "Programming", "Software Engineering Components" of the curriculum of bachelor's training in the specialty 121 Engineering Software.

Theoretical knowledge and practical skills obtained during the study of the discipline "Mobile Application Development" ensure the successful completion of course projects and diploma projects in the specialty 121 Software Engineering.

3. Content of the academic discipline

The discipline "Mobile Application Development" involves the study of the following topics:

Topic 1. Basic concepts of the Android OS

Topic 2. Creating a user interface

Topic 3. Databases and data sources

Topic 4. Sensors

Modular control work

Credit

4. Educational materials and resources

Basic literature:

1. *Beck K. Test-Driven Development: By Example / K. Beck. - Addison-Wesley Longman, 2002. – 240 p.*
2. *Clark J. Designing for Touch / J. Clark. - 2015. - 169 p.*
3. *Griffiths D. Head First Android Development / D. Griffiths, D. Griffiths. - O'Reilly Media, 2015. - 734p.*
4. *Keith J. HTML5 for Web Designers / J. Keith, R. Andrew. - 2016. - 92 p.*
5. *Marcotte E. Responsive Web Design / E. Marcotte. - 2014. - 153 p.*
6. *McGrane K. Content Strategy for Mobile / K. McGrane. - 2012. - 165p.*

Familiarize yourself with the sections related to the following topics of the discipline: basic concepts of the Android OS, creating a user interface, databases and data sources, sensors. The materials are freely available on the Internet.

Additional literature:

3. CREATE YOUR FIRST APP FOR ANDROID: http://mikrotik.kpi.ua/index.php/courses-list/android/39-create-your-first-app-for-android.

4. Panigrahy N. Xamarin Mobile Application Development for Android /Nilanchala Panigrahy. - Packt Publishing, 2015. – 296p.

5. Petzold C. Creating Mobile Apps with Xamarin.Forms. – WWW: https://developer.xamarin.com/guides/xamarin-forms/creating-mobile-apps-xamarin-forms/

6. Schwarz R. The Android Developer's Cookbook: Building Applications with the Android SDK / R. Schwarz, P. Dutson, J. Steele, N. To. - Addison-Wesley, 2013. - 464p.

7. Troelsen A. Pro C# 2010 and the .NET 4.0 Platform / A. Troelsen. ¬ WWW: http://www.apress.com/book/view/9781430225492

8. Wroblewski L. Mobile First / L. Wroblewski. - 2011. - 123 p.

9. The Java® Language Specification [Electronic resource]. – 2019. – Resource access mode: https://docs.oracle.com/javase/specs/jls/se12/html/index.html

10. Kotlin. https://kotlinlang.org/

11. Android developer guides. https://developer.android.com/guide

Use to master the practical skills of the discipline. The materials are freely available on the Internet.

Educational content

5. Methods of mastering an educational discipline (educational component)

No. z/p	*Type of training session: Lecture*	*Description of the training session*
	Topic 1. Basic concepts of the Android OS	
1	*Lecture 1. Development of mobile operating systems.*	*Features of development for mobile operating systems. The history of mobile OS development. Overview of modern mobile operating systems. Overview of programming languages for mobile platforms* *Task for self-training: item 6 #1.*
2	*Computer workshop 1*	*Task: Implementation of the "Traffic light" application* *Task for self-training: item 6 #2.*
3	*Lecture 2. Android platform architecture.*	*Architecture of the Android platform. Android application components. Dalvik Java Virtual Machine. Android SDK versions and compatibility* *Task for self-training: item 6 #3.*
4	*Lecture 3. Basic concepts of the Android OS.*	*Android OS. Basic concepts. Integrated development environments. Project structure in Android Studio. Android emulators. Configuration of emulators. Debugging the application* *Task for self-training: item 6 #4.*

5	Computer workshop 2	Task: create an application with several activities and provide transition between windows (from the main window to another and back) Task for self-training: item 6 #5.
6	Lecture 4. The process of building Android applications.	Application manifest file. The process of building Android applications. The Activity component of the Android platform and the application life cycle. Registration of Activity life cycle events. Task for self-training: item. 6 # 6.
Topic 2. Creating a user interface		
7	Lecture 5. Basics of creating a user interface.	The basics of creating a user interface. Building a layout of the user interface. Resources and resource identifiers. Layout managers. Task for self-training: item 6 #7.
8	Computer workshop 3	Task: Creating a menu Task for self-training: item 6 #8.
9	Lecture 6. Adapters.	Adapters. Connecting widgets and assigning listeners. UI fragments. Life cycle of fragments. Hosting of UI fragments. FragmentManager and fragment transactions. Task for self-training: item 6 #9.
10	Lecture 7. Styles. Topics Menu.	Styles. Topics Creating a menu. ViewPager and dividing the program into pages. TimePickerDialog. DialogFragment and creation of dialog boxes. Transferring data to a dialog box. Task for self-training: item 6 #10.
11	Computer workshop 4	Task: Implementation of background streams. Task for self-training: item 6 #11.
12	Lecture 8. Work in the background mode.	Work in the background. Creation, launch and management of the service. Connecting services with activities. Creation of background services. Task for self-training: item 6 #12.
13	Lecture 9. Use of background streams.	Using background threads. Alarms. Using AsyncTask to run asynchronous tasks. IntentService. Using Looper, Handler and HandlerThread Task for self-training: item 6 #13.
14	Computer workshop 5	Task: Implementation of asynchronous tasks. Task for self-training: item 6 #14.
Topic 3. Databases and data sources		
15	Lecture 10. Databases and data sources.	Databases and data sources. Working with SQLite databases. Cursors and the ContentValues class.

		Using SQLiteOpenHelper. Execution of requests to the database. Task for self-training: item 6 #15.
16	Lecture 11. Creation and registration of data sources.	Creation and registration of data sources. Asynchronous requests to data sources using CursorLoader. Search implementation. Standard data sources in Android (MediaStore, ContactsContract, Calendar). Task for self-training: item 6 #16.
17	Computer workshop 6	Task: Work with SQLite database in Android Task for self-training: item 6 #17.
18	Lecture 12. Maps, geolocation.	Maps, geocoding and geolocation services. Use of geolocation services LocationManager and LocationProvider. Search for geolocation data sources according to the specified criteria. Find your current location Task for self-training: item 6 #18.
19	Lecture 13. Creation of interactive maps.	Using proximity notifications. Reverse and forward geocoding. Creating interactive maps using MapView and MapActivity. Using the MapController object. Layering. Task for self-training: item 6 #19.
20	Computer workshop 7	Task: Adding and deleting rows in SQLite database, grouping, sorting, selection of data by condition. Task for self-training: item 6 #20.
Topic 3. Sensors		
21	Lecture 14. Sensors.	Sensors. Low-level Android programming interfaces. Using the SensorManager object. Tracking sensor readings. Task for self-training: item 6 #21.
22	Lecture 15. Management of Bluetooth devices.	Management of Bluetooth devices and discovery mode. Detection of remote Bluetooth devices. Interaction via Bluetooth. Internet connection tracking. Task for self-training: item 6 #22.
23	Computer workshop 8	Task: Queries from linked tables to the SQLite database. Transactions. Task for self-training: item 6 #23.
24	Lecture 16. Tracking information about Wi-Fi and other networks.	Tracking information about Wi-Fi and other networks. Changing Wi-Fi configuration and finding access points. Data transfer using Wi-Fi Direct. Scanning of NFC tags. Data transfer using Android Beam technology.

		Task for self-training: item 6 #24.
25	*Lecture 17. Modern technologies and development methods on Kotlin.*	*Modern technologies and methods of Kotlin development. Interaction of software with the JS language.* *Task for self-training: item 6 #25.*
26	*Lecture 18. Support and further development of mobile applications.*	*Support and further development of mobile applications. Software testing and documentation. Implementation of developed mobile applications. Functionality update and software promotion.* *Task for self-training: item 6 #26.*
Modular control work		

6. Independent work of a student/graduate student

The discipline "Mobile Applications Development" is based on independent preparations for classroom classes on theoretical and practical topics.

No. z/p	*The name of the topic submitted for independent processing*	*Number of hours*	*literature*
1	*Preparation for the lecture 1*	*1*	*1; 2; 3;*
2	*Preparation for the computer workshop 1*	*3*	*1; 2; 3;*
3	*Preparation for lecture 2*	*1*	*1; 2; 3;*
4	*Preparation for the lecture 3*	*1*	*1; 2; 3;*
5	*Preparation for the computer workshop 2*	*3*	*1; 2; 3;*
6	*Preparation for the lecture 4*	*1*	*1; 2; 3;*
7	*Preparation for the lecture 5*	*1*	*1; 2; 3;*
8	*Preparation for the computer workshop 3*	*3*	*1; 2; 3;*
9	*Preparation for the lecture 6*	*1*	*1; 2; 3;*
10	*Preparation for the lecture 7*	*1*	*1; 2; 3;*
11	*Preparation for the computer workshop 4*	*3*	*1; 2; 3;*
12	*Preparation for the lecture 8*	*1*	*1; 2; 3;*
13	*Preparation for the lecture 9*	*1*	*1; 2; 3*
14	*Preparation for the computer workshop 5*	*3*	*1; 2; 3;*
15	*Preparation for lecture 10*	*1*	*1; 2; 3;*
16	*Preparation for lecture 11*	*1*	*1; 2; 3;*
17	*Preparation for the computer workshop 6*	*3*	*1; 2; 3;*
18	*Preparation for lecture 12*	*1*	*1; 2; 3;*
19	*Preparation for lecture 13*	*1*	*1; 2; 3;*
20	*Preparation for the computer workshop 7*	*3*	*1; 2; 3;*
21	*Preparation for lecture 14*	*1*	*1; 2; 3;*

22	Preparation for lecture 15	1	1; 2; 3;
23	Preparation for the computer workshop 8	3	1; 2; 3;
24	Preparation for lecture 16	1	1; 2; 3;
25	Preparation for lecture 17	1	1; 2; 3;
26	Preparation for lecture 18	1	1; 2; 3;
27	Preparation for modular control work	12	1-3
28	Preparation for the test	12	1-3

Policy and control

7. Policy of academic discipline (educational component)

Attending lectures is mandatory.

Attending computer workshop classes may be occasional and for consultation/protection of computer workshop works.

Rules of behavior in classes: activity, respect for those present, turning off phones.

Adherence to the policy of academic integrity.

Rules for protecting the work of the computer workshop: the work must be done in accordance with the tasks and according to the option.

The rules for assigning incentive and penalty points are as follows. Incentive points are awarded for:
- accurate and complete answers in surveys based on lecture materials (maximum number of points per survey - 3 points).

8. Types of control and rating system for evaluating learning outcomes (RSO)

During the semester, students perform 8 computer workshops. The maximum number of points for each computer workshop: 5 points.

Points are awarded for:
- quality of performance of the computer workshop: 0-3 points;
- answer during the defense of the computer workshop: 0-2 points.

Performance evaluation criteria:
3 points - the work is done with quality, in full;
2 points – the work is done qualitatively, in full, but has flaws;
1 points – the work is completed in full, but contains significant errors;
0 points - the work is not completed in full.

Answer evaluation criteria:
2 points – the answer is complete, well-argued;
1 points – the answer is correct, but has flaws or minor errors;
0 points - there is no answer or the answer is incorrect.

The maximum number of points for performing and defending computer practicals:
5 points × 8 comp. practice = 40 points.

During the semester, lectures take place on the topic of the current lesson. Maximum points for all surveys: 3 points. The number of surveys on the topic of the current lesson for one student is unlimited.

The task for the modular control work consists of 2 theoretical and 2 practical tasks on the topics "Designing mobile subprograms using Java" and "Development of software for Android mobile *applications using Android Studio". The answer to each theoretical question is evaluated with 15 points, the implementation of each practical task is evaluated with 15 points.*

Evaluation criteria for the theoretical test question:
13-15 points – the answer is correct, complete, well-argued;
10-12 points – the answer is correct, detailed, but not very well argued;
7-9 points - in general, the answer is correct, but has shortcomings;
4-6 points – there are minor errors in the answer;
1-3 points – there are significant errors in the answer;
0 points - there is no answer or the answer is incorrect.

Evaluation criteria for the practical task of the control work:
12-15 points - the task was completed qualitatively, in full;
8-11 points - the task was completed with minor shortcomings;
3-7 points – the task is completed in full, but contains significant errors;
0-2 points – the task was not completed in full.

The maximum number of points for a modular control work:
15 points × 2 questions + 15 points × 2 tasks = 60 points.

The rating scale for the discipline is equal to:
R = RS = 40 points + 60 points = 100 points.

Calendar control: is conducted twice a semester as a monitoring of the current state of fulfillment of the syllabus requirements.
At the first certification (8th week), the student receives "credited" if his current rating is at least 10 points (50% of the maximum number of points a student can receive before the first certification).
At the second certification (14th week), the student receives "passed" if his current rating is at least 17 points (50% of the maximum number of points a student can receive before the second certification).

Semester control: assessment
Conditions for admission to semester control:
With a semester rating (RC) of at least 60 points and the completion of all computer lab work, the student receives credit "automatically" according to the table (Table of correspondence of rating points to grades on the university scale). Otherwise, he has to perform the final control work.

Completion and defense of the computer workshop is a necessary condition for admission to the credit control work.

If the student does not agree with the "automatic" grade, he can try to improve his grade by writing a credit test, while his points received for the semester are kept, and the better of the two grades received by the student is assigned ("soft" grading system) .

Table of correspondence of rating points to grades on the university scale:

Scores	*Rating*
100-95	Perfectly
94-85	Very good
84-75	Fine
74-65	Satisfactorily
64-60	Enough
Less than 60	Unsatisfactorily
Admission conditions not met	Not allowed

9. Additional information on the discipline (educational component)

The list of questions submitted for semester control is given in Appendix 1.

Working program of the academic discipline (syllabus):

Folded assistant, K. O. Radchenko

Adopted by Computer Systems Software Department (protocol № 8 from 25.01.23)

Approved by the Faculty Board of Methodology (protocol № 6 from 27.01.23)

Appendix 1. List of questions submitted for semester control

1. *Android OS architecture. Dalvik Java Virtual Machine.*
2. *Principles of creating Android applications. Application components.*
3. *Application manifest.*
4. *Resources and resource identifiers. Layout managers.*
5. *Activities. Activity life cycle.*
6. *Fragments. Life cycle of fragments. Hosting of UI fragments.*
7. *Intentions. Explicit and implicit intentions. Intent filters.*
8. *Broadcast receivers and methods of their registration.*
9. *Creation, launch and management of the service.*
10. *Connecting services with activities.*
11. *Working with SQLite databases.*
12. *Cursors and the ContentValues class.*
13. *Concept of content provider. Access to the provider. Provider URI. Receiving data from the provider.*
14. *Concept of process. Process life cycle. Types of processes.*
15. *Using AsyncTask to launch asynchronous tasks. Interaction between processes.*
16. *Use of geolocation services LocationManager and LocationProvider.*
17. *Reverse and forward geocoding.*
18. *Creation of interactive maps using MapView and MapActivity.*
19. *Using the SensorManager object.*
20. *Management of Bluetooth devices and discovery mode.*