

МІНІСТЕРСТВО ОСВІТИ Й НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
КАФЕДРА ПРОГРАМНИХ ЗАСОБІВ

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

до виконання курсового проекту з дисципліни
«РОЗРОБКА ЗАСТОСУВАНЬ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ»

*для студентів за напрямом підготовки 6.050101
"Комп'ютерні науки"*

усіх форм навчання

Запоріжжя 2016

Методичні рекомендації до виконання курсового проекту з дисципліни “РОЗРОБКА ЗАСТОСУВАНЬ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ” для студентів напрямку **6.050101 "Комп'ютерні науки"**. / Укл.: О.О. Степаненко – Запоріжжя: ЗНТУ, 2016. – 29 с.

Укладачі: О.О. Степаненко, к.т.н., доцент кафедри ПЗ,

Рецензент: В.Ф. Онищенко, к.ф.-м.н., доцент кафедри ПЗ.

Відповідальний

за випуск: С.О. Субботін, зав. каф. ПЗ, д.т.н., проф.

Затверджено
на засіданні кафедри
“Програмних засобів”

Протокол №
від "___" _____ 2016р.

ЗМІСТ

ВСТУП	4
1 ЗМІСТ КУРСОВОГО ПРОЕКТУ	5
1.1 ЗМІСТ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ.....	5.
1.2 ЗМІСТ ГРАФІЧНОЇ ЧАСТИНИ.....	10
2 ОСНОВНІ ЕТАПИ РОЗРОБКИ КУРСОВОГО ПРОЕКТУ	12
3 ОРГАНІЗАЦІЯ ПРОЕКТУВАННЯ І ПОРЯДОК ЗАХИСТУ	
КУРСОВОГО ПРОЕКТУ.....	14
4 ПЕРЕЛІК МОЖЛИВИХ ТЕМ КУРСОВОГО ПРОЕКТУ.....	16
5 ПЕРЕЛІК ПОСИЛАНЬ.....	19
ДОДАТОК А ТИТУЛЬНИЙ ЛИСТ ПОЯСНЮВАЛЬНОЇ	
ЗАПИСКИ....ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.	
ДОДАТОК Б БЛОК-СХЕМА ПРОГРАМИ (ПРИКЛАД)	23
ДОДАТОК В ТЕКСТ МОБІЛЬНОГО ДОДАТКУ.....	25

ВСТУП

Розробка додатків та ігор для мобільних пристроїв — це сфера сучасних технологій, що найбільш активно розвивається. Мобільні пристрої окрім традиційних об'єднують в собі велику кількість різноманітних функцій, для яких раніше використовувались інші пристрої. На фоні глобальної мобілізації, активного росту ринку мобільних додатків, все більш актуальним стає питання про розробку мобільних додатків.

Метою курсового проекту є вироблення у студентів самостійних навиків аналізу тематики, вибору платформи розробки застосування, розробки архітектурної моделі, складання алгоритму і програмування на мовах, сумісних з вибраною платформою. Студент в процесі проектування самостійно, на підставі знань і практичних навичок, отриманих ним при вивченні курсу «Розробка застосувань для мобільних пристроїв», обирає платформу, розробляє додаток або програмну систему, використовуючи основні можливості мов Java, Objective-C, C/C++ бібліотеки, C# та інш.

Виконання курсового проекту готує студента до самостійної розробки програм та застосувань, що мають практичну та навчальну цінність, як інструменту при вивченні інших предметів та під час професійної підготовки, а також як можливість реалізації основних розділів випускної або дипломної роботи.

В процесі роботи над курсовим проектом студент набуває навичок науково-дослідної і самостійної творчої роботи. При виконанні проекту студенти отримують досвід роботи з довідковою літературою, стандартами, а також навички складання і оформлення програмної документації.

Тематику, основні положення проекту, оформлення пояснювальної записки студенти погоджують з керівником проекту у міру готовності відповідного матеріалу під час консультацій.

1 ЗМІСТ КУРСОВОГО ПРОЕКТУ

До складу курсового проекту входить:

1. Пояснювальна записка обсягом 30 - 40 листів (не включаючи додатків).
2. Презентація програмної розробки у вигляді *.ppt файлу.
3. Програмна розробка (застосування) на мобільному носії інформації.

1.1 Зміст пояснювальної записки

Пояснювальна записка оформляється на листах білого паперу формату А4, зшитих зліва. Пояснювальна записка повинна бути роздрукована комп'ютерним способом з дотриманням стандартів для текстових документів (шрифт Times New Roman №14, з полуторним інтервалом).

Пояснювальна записка повинна містити наступні розділи:

Титульний лист.

Завдання на курсовий проект.

Реферат.

Перелік умовних позначень, символів, одиниць і термінів.

Зміст.

Вступ.

- 1 Аналіз предметної області.
- 2 Технічне завдання на розробку.
- 3 Вибір і обґрунтування структури системи, що проектується.
- 4 Керівництво програміста.
- 5 Керівництво оператора (або користувача).
- 6 Тестування розробки.

Висновки.

Перелік посилань.

Додаток А Текст програми

Додаток Б Слайди презентації (8 листів), або 2 листа формату А3 (блок-схема та логічна схема).

Приклад оформлення титульного листа приведений у Додатку А цих методичних рекомендацій.

Текст пояснювальної записки ділять на розділи. Кожен розділ, як правило, починається з нової сторінки. Розділи в межах всієї пояснювальної записки, а також підрозділи і пункти мають порядкові номери, позначені арабськими цифрами без крапки. Вступ і висновок (виводи) не нумеруються.

Заголовки розділів друкують посередині тексту. Заголовки підрозділів пишуть з абзацу, відступаючи зліва 15 мм. У заголовку не допускаються перенесення слів. Крапку в кінці заголовка не ставлять.

Нумерація сторінок пояснювальної записки - наскрізна. Номер сторінки проставляється арабськими цифрами в середині верхнього поля сторінки. На титульному листі номер 1 не ставлять.

Завдання на курсовий проект оформляється на бланку встановленої форми.

Реферат (скорочений виклад пояснювальної записки з основними відомостями і висновками) будується за схемою:

а) відомості про обсяг пояснювальної записки, кількості ілюстрацій, таблиць, схем, використаних джерел;

б) текст реферату.

Текст реферату включає такі відомості: предмет, характер і мета роботи; метод проведення роботи; тип використаного комп'ютера; мова програмування (інтегроване середовище), висновки про результати (особливості, ефективність, можливості);

в) ключові слова.

Об'єм реферату - не більше 1 сторінки.

Зміст включає номери і найменування всіх розділів, підрозділів і додатків з вказівкою номерів сторінок, на яких розміщується початок матеріалу.

Перелік умовних позначень, символів, одиниць і термінів. Дається короткий перелік всіх умовних позначень, символів, одиниць і термінів, що використовуються в пояснювальній записці в алфавітному порядку (спочатку кирилиця, потім латиниця).

Вступ повинний відображати сучасний стан проблеми, що розглядається в курсовому проекті, її актуальність. Розмір цієї частини пояснювальної записки не повинен перевищувати 1-2 сторінки.

У розділі «**Аналіз предметної області**» слід зробити загальний огляд та огляд існуючих або можливих методів вирішення поставленої задачі і обґрунтувати необхідність або доцільність вибраного студентом нового методу рішення або нової реалізації існуючого методу. Необхідно також представити архітектурну або структурну модель (у випадку, якщо це необхідно).

Розділ «**Технічне завдання на розробку**» повинен містити тему, мету та основні теоретичні посилання на розробку, вимоги до додатку, докладний словесний опис узагальненого алгоритму функціонування додатку в термінах предметної області (наприклад, у вигляді: «Крок 1:.; Крок 2: . і т.д.) При описі алгоритму не слід використовувати конкретні імена змінних, функцій і т.п..

Наповнення розділу «**Вибір і обґрунтування структури системи, що проектується**» наступне: обґрунтований вибір платформи та операційної системи, на якій планується функціонування додатку, вибір інтегрованого середовища розробки та/або мови програмування, обґрунтування структури системи, обрання тих чи інших компонентів програми та середовища.

При оформленні розділу «**Керівництво програміста**» потрібно звернути увагу на такі підрозділи як: загальні відомості про додаток, призначення і умови застосування, його основні характеристики (часові характеристики, режим роботи, засоби контролю правильності виконання і т.п.), логічна та

функціональна структура додатку, необхідне апаратне та програмне забезпечення, технології, методи та класи, що використовуються, змінні та константи, повідомлення та інші.

Подається опис логічної структури додатку - її склад (якщо вона складається з декількох модулів (файлів)), функції, що входять до складу додатку, їх взаємозв'язок, спосіб обміну параметрами між функціями, системні функції та модулі і т.д. Необхідно зробити ув'язку даного додатку з іншими складовими архітектури платформи. У цьому ж підрозділі вказується тип мобільного пристрою і приводиться перелік інших необхідних пристроїв, вживаних при роботі додатку. Вказується спосіб звернення до додатку, виклик з носія, відомості про використання оперативної пам'яті, обсяг програмного додатку.

Обов'язковим є наявність пункту **про розробника** в інтерфейсі користувача у додатку.

До складу розділу **«Керівництво оператора (користувача)»** входять такі відомості: призначення додатку, його функції і особливості експлуатації, умови виконання додатку, які пов'язані з складом апаратних і програмних засобів, виконання додатку (зведення про послідовність дій оператора, що забезпечують завантаження, запуск і завершення додатку, а також опис функцій, формату і можливих варіантів команд, за допомогою яких оператором (користувачем) повинні здійснюватися завантаження додатку і управління його виконанням, а також відповіді на ці команди).

Важливим також є заключний розділ пояснювальної записки **«Тестування розробки»**. В ньому подаються етапи функціонування додатку, можливі варіації вхідних та вихідних даних, особливості застосування додатку, можливі помилки у роботі додатку, що були виявлені на тих чи інших етапах функціонування, або їх відсутність. Робиться висновок про якісні та ефективні характеристики роботи додатку.

При оформленні та наповненні усіх розділів стосовно розробки програмного застосування необхідно користуватися програмними документами, відповідно вимогам стандарту ЄСПД (Єдина Система Програмної Документації) та ДСТУ 3918-1999 (ISO/IEC 12207:1995) «Види програм і програмних документів», - «Опис програми», «Керівництво програміста», «Керівництво оператора».

Наприкінці повинні бути зроблені **висновки** за наслідками виконання курсового проекту, проаналізовані достоїнства розробки, відмічені наявні недоліки і шляхи їх усунення, а також способи вдосконалення розробленого додатку.

Перелік посилань повинен містити перелік джерел (книг, статей, методичних вказівок, посилань на ресурси глобальної мережі і т.п.), використаних при виконанні проекту. Джерела розташовуються в порядку появи посилань в тексті пояснювальної записки. Посилання в тексті пояснювальної записки подаються в квадратних скобках [].

Текст пояснювальної записки повинен бути від третьої особи. Стиль пояснювальної записки - строгий стиль технічного документа.

Додатки можуть містити код, ілюстративний матеріал, таблиці, текст допоміжного характеру. Додатки оформлюють як продовження пояснювальної записки на подальших її сторінках, розташовуючи їх в порядку появи посилань в тексті. Кожний додаток слід починати з нової сторінки.

Додаток повинний мати заголовок, надрукований зверху в центрі сторінки. Над заголовком повинне бути надруковане слово «Додаток ____» і велика буква, яка позначає додаток.

Додатки позначають послідовно великими буквами українського алфавіту, виключаючи Г Є, З І, Ї, Й, О, Ч, Ь, наприклад, Додаток А, Додаток Б, і т.д. Додатки повинні мати загальну з рештою записки наскрізну нумерацію сторінок.

В обов'язкових додатках повинні бути приведені програмні документи «Текст програми» ДСТУ 3918-1999 (ISO/IEC 12207:1995) та слайди презентації. Текст програми повинен бути твердою копією (лістингом) відлагодженої та правильно функціонуючої програми, яка реалізує рішення поставленої задачі на проектування. Програма повинна бути забезпечена достатньою (але не надмірною) кількістю коментарів і містити не менше 200 виконуваних операторів мови. Обов'язковим є розбиття програми додатку на функції та методи (модулі), які обмінюються інформацією, отримуючи параметри і повертаючи значення. Як правило, програма повинна використовувати зовнішні файли як джерело інформації або як місце знаходження результатів роботи. Програма повинна мати добре розроблений та продуманий інтерфейс з користувачем, виводити достатню кількість повідомлень. При необхідності введення початкових даних повинен здійснюватися контроль за правильністю введення.

Склад і зміст документів курсового проекту у кожному конкретному випадку узгоджуються з керівником і може відрізнятися від розглянутих у вказівках.

1.2 Зміст графічної частини

Графічна частина виконується як обов'язковий додаток Б до пояснювальної записки у вигляді слайдів презентації, але за узгодженням з керівником може містити 2 листи формату А3. У них повинні бути представлені схема алгоритму, оформлена згідно ГОСТ ЄСПД 19.701-90, і структурна (логічна або функціональна) схема програми або програмної системи (додатку).

Якщо графічна частина виконується на листах А3, схеми можуть бути виконані олівцем або тушшю, або комп'ютерним способом з дотриманням стандартів (див. Додаток Б). На листах повинен бути присутнім основний напис.

ГОСТ 19.701-90 розповсюджується на умовні графічні позначення (символи) в схемах алгоритмів і програм, що відображають основні операції процесу обробки даних і програмування для систем програмного забезпечення обчислювальних машин, комплексів і систем незалежно від їх призначення і області застосування.

У випадку, якщо студент вирішив в якості **додатку Б** оформити слайди презентації (не менше 8 слайдів формату А4), необхідно також продублювати матеріали презентації на флеш-записуючому пристрої (ppt файл). В презентації обов'язковими є наступні слайди: назва розробки та повна інформація про розробника; тема, мета і задачі розробки; вибір платформи, середовища розробки, мови; логічна/функціональна схема; знімок (знімки) екрану роботи додатку; висновки.

2 ОСНОВНІ ЕТАПИ РОЗРОБКИ КУРСОВОГО ПРОЕКТУ

Стандартом ГОСТ 19.102 -77 ЄСПД встановлюються стадії розробки програм і програмної документації для обчислювальних машин, комплексів і систем незалежно від їх призначення і області застосування. Згідно цьому стандарту, стадії розробки, етапи і зміст робіт повинні відповідати вказаним в табл. 2.1.

Таблиця 2.1 - Стадії розробки програм і програмної документації.

Стадії розробки	Етапи робіт	Зміст робіт
1	2	3
1. Технічне завдання	Обґрунтування необхідності розробки програми	Постановка завдання. Збір вихідних матеріалів. Вибір і обґрунтування критеріїв ефективності і якості розроблювальної програми. Обґрунтування необхідності проведення науково-дослідних робіт.
	Науково-дослідні роботи	Визначення структури вхідних і вихідних даних. Попередній вибір методів рішення задач. Обґрунтування доцільності застосування раніше розроблених програм. Визначення вимог до технічних засобів. Обґрунтування принципової можливості рішення поставленої задачі.
	Розробка й затвердження технічного завдання	Визначення вимог до програми. Розробка техніко-економічного обґрунтування розробки програми. Визначення стадій, етапів і строків розробки програми й документації на неї. Вибір мов програмування. Визначення необхідності проведення науково-дослідних робіт на наступних стадіях. Узгодження й затвердження технічного завдання.
2. Ескізний проект	Розробка ескізного проекту	Попередня розробка структури вхідних і вихідних даних. Уточнення методів рішення задачі. Розробка загального опису алгоритму рішення задачі. Розробка техніко-економічного обґрунтування.
	Затвердження ескізного проекту	Розробка пояснювальної записки. Узгодження й затвердження ескізного проекту.

Продовження табл. 2.1

1	2	3
3.Технічний проект	Розробка технічного проекту	Уточнення структури вхідних і вихідних даних. Розробка алгоритму рішення задачі. Визначення форми подання вхідних і вихідних даних. Визначення семантики й синтаксису мови. Розробка структури програми. Остаточне визначення конфігурації технічних засобів.
	Затвердження технічного проекту	Розробка плану заходів щодо розробки й впровадження програми. Розробка пояснювальної записки. Узгодження й затвердження технічного проекту.
4.Робочий проект	Розробка програми	Програмування й налагодження програми. Виготовлення програми-оригіналу.
	Розробка програмної документації	Розробка програмних документів відповідно до вимог ГОСТ
	Випробування програми	Розробка, узгодження й затвердження порядку й методики випробувань. Проведення попередніх, державних, міжвідомчих, приймально-здавальних і інших видів випробувань Коректування програми й програмної документації за результатами випробувань
5.Впровадження	Підготовка й передача програми	Підготовка й передача програми й програмної документації для супроводу й (або) виготовлення. Оформлення й затвердження акту про передачу програми на супровід і (або) виготовлення. Передача програми до фонду алгоритмів і програм.

При складанні плану виконання курсового проекту слід враховувати цей стандарт.

3 ОРГАНІЗАЦІЯ ПРОЕКТУВАННЯ І ПОРЯДОК ЗАХИСТУ КУРСОВОГО ПРОЕКТУ

На другому тижні семестру студенти отримують у керівника завдання на курсовий проект. Тема курсового проекту може бути запропонована студентом на підставі матеріалів, зібраних під час учбової практики, і представлена викладачеві. Керівник затверджує тему (можливо, із змінами) або пропонує тему з числа учбових (перелік учбових тем представлений в розділі 5 цих методичних вказівок).

Студент заповнює бланк завдання на курсове проектування і календарний план виконання проекту.

Керівник роз'яснює вимоги і дає загальні рекомендації по виконанню курсового проекту, рекомендує загальну і спеціальну літературу, допомагає вирішити складні технічні питання.

За якість виконання проекту і правильність ухвалених рішень, а також за своєчасне завершення роботи відповідає студент. При виборі тих або інших рішень ініціатива надається студентові. Окрім технічного керівництва, основним завданням керівника є прищеплення студентові навичок творчої і систематичної самостійної роботи.

Протягом семестру студент відвідує консультації по курсовому проектуванню і пред'являє керівникові на перевірку матеріали відповідно до календарного плану і графіка виконання курсового проекту.

Проект представляється до захисту не пізніше встановленого керівником терміну. Заздалегідь курсовий проект в повному об'ємі пред'являється керівникові і підписується ним після усунення студентом всіх зауважень.

Захист курсових проектів відбувається відповідно до графіка, складеного на кафедрі. Для прийому курсових проектів створюється комісія у складі керівника проекту і 1-2 викладачів кафедри.

На захисті студент повинен продемонструвати роботу програмного додатку, представити правильно оформлену пояснювальну записку і графічну частину. Студент повинен уміти відповісти на питання, пов'язані з тематикою роботи, розробленим алгоритмом (функціональною або структурною схемами), методами та технологіями, що використані ним у роботі.

4 ПЕРЕЛІК МОЖЛИВИХ ТЕМ КУРСОВОГО ПРОЕКТУ

Курсовий проект з дисципліни «Розробка застосувань для мобільних пристроїв» виконується за одним з напрямів або тем:

1. Розробка додатків-довідників і інформаційних програмних систем.
2. Розробка додатків-імітаторів і ігрових програм.
3. Розробка текстових і графічних редакторів.
4. Розробка додатків для вирішення фінансово-економічних задач.
5. Розробка ігрового додатку «Життя». (Задане прямокутне поле комірок, кожна з яких має 8 сусідів і може містити «організм»; $m(k)$ - кількість сусідніх з k -ю коміркою комірок, зайнятих «організмами». Нове покоління «організмів» походить із попереднього покоління за правилами:

- «організм» в комірці k переходить у наступне покоління за умови $2 \leq m(k) \leq 3$, інакше він умирає;
- «організм» народжується в порожній комірці k при $m(k)=3$.

Програма повинна генерувати початковий стан поля комірок заданої величини, відображати його на екрані, і демонструвати зміну поколінь по натисканні клавіші або після закінчення заданого проміжку часу.)

6. Розробка додатку - калькулятора.
7. Розробка додатку для шифрування/дешифрування файлу.
8. Розробка додатку для архівації/розархівації файлів.
9. Розробка додатку для статистичного аналізу текстового файлу. (У заданому вхідному файлі програма повинна підраховувати кількість слів, що складаються з 1-ї, 2-х, 3-х і т.д. букв, і виводити на екран гістограму довжин цих слів).
10. Розробка інформаційно-довідкової системи «Телефонна книга».
11. Розробка ігрового додатку «Морський бій».
12. Розробка ігрового додатку «Хрестики-Нулики».

13. Розробка додатку пошуку проходу по лабіринту. (Лабіринт - двовимірний масив, заповнений 1 і 0 випадковим образом, має 1 вхід (постійний) і 1 вихід (випадковий). Один з алгоритмів пошуку проходу - рухатися вперед, тримаючись правою рукою за стіну. При повороті стіни треба також повертатися. Якщо вихід можливий, він буде знайдений, у протилежному випадку буде знайдений вхід.)
14. Розробка додатку для генерації текстів. (У програмі повинні використовуватися 3 файли, у яких зберігаються 1) іменники; 2) дієслова; 3) прийменники. Випадковим образом з файлів повинні бути обрані слова, з яких складаються речення виду: «Іменник дієслово прийменник іменник», які записуються в проміжний масив, а потім у результуючий файл. Слова повинні відділятися друг від друга пробілами. Речення починається із заголовної букви й закінчується крапкою (можливо, знаком оклику або знаком питання). Створити файл, що складається з 20 таких речень. Передбачити можливість уведення нових слів і запам'ятовування їх у файлах.)
15. Розробка додатку для кодування/декодування тексту за допомогою абетки Морзе.
16. Розробка додатку MP3 плеєра.
17. Розробка додатку відео-плеєра.
18. Розробка додатку читання книжок (book-reader).
19. Розробка додатку з обміну файлами по технології Bluetooth.
20. Розробка додатку з обміну файлами по технології Wi-Fi.
21. Розробка додатку диктофону.
22. Розробка додатку багатофункціонального будильника.
23. Розробка додатку календаря-нагадувальника.
24. Розробка додатку словника-перекладача.
25. Програма-кулінар з підрахунком калорій.
26. Мобільний додаток «Журнал погоди».

27. Мобільний додаток «Записна книжка» з нагадувачем.

При погодженні з керівником проекту тема може бути іншою, ніж є в списку.

5 ПЕРЕЛІК ПОСИЛАНЬ

1. И.Салмре Программирование мобильных устройств на платформе .NET Compact Framework. “Вильямс”, 2006. - 736 с.
2. Голощапов А. Л. Google Android: программирование для мобильных устройств / Голощапов А. Л. – СПб. : БХВ-Петербург, 2011. – 448 с. : ил. + CD-ROM – (Профессиональное программирование).
3. Р.Майер. Android 2: программирование приложений для планшетных компьютеров и смартфонов. – М.: Эксмо, 2011. – 672 стор.
4. Горнаков С.Г. Программирование мобильных телефонов на Java 2 Micro Edition / С.Г. Горнаков. – М.: ДМК Пресс, 2004. – 336 с.
5. Пирумян В.В. Платформа программирования J2ME для портативных устройств / В.В. Пирумян Пер. с англ. – М.: КУДИЦ-ОБРАЗ, 2002. – 352 с.
6. Архипкин В.Я. Bluetooth. Технические требования. Практическая реализация / В.Я. Архипкин, А.В. Архипкин. – М.: Мобильные коммуникации, 2004. – 203 с.
7. Дэрс Л. Android за 24 часа. Программирование приложений под операционную систему Google / Дэрс Л., Кондер Ш. – М. : Рид Групп, – 464 с.
8. Майер Р. Android 2: программирование для планшетных компьютеров и смартфонов / Ретро Майер ; пер. с англ.– М. : Эксмо, 2011. – 672 с.
9. Офіційний блог компанії Google
<http://googleblog.blogspot.com/search/label/Android>.
- 10.Онлайн-підтримка StackOverflow
<http://stackoverflow.com/questions/tagged/android>.
11. Альянс відкритих мобільних пристроїв, <http://www.openhandsetalliance.com/>.
12. Хабрахабр – Все для программиста [Электронный ресурс]. – Режим доступа : <http://habrahabr.ru/>.

13. Исследование «Рынок разработки мобильных приложений» [Электронный ресурс] – режим доступа до статті:
<http://adindex.ru/publication/mediaoutlook/digital/2013/08/13/101216.phtml>
14. Мобильные игры – основне участники и перспективы отрасли [Электронный ресурс] – режим доступа до статті:
<http://www.3dnews.ru/572338>
15. Категории мобильных приложений и частота их использования [Электронный ресурс] – режим доступа до статті:
<http://www.cossa.ru/articles/149/27855/>
16. Троелсен Е. С# и платформа .NET. Библиотека программиста. – СПб.: Питер, 2007. – 796 с.
17. Обзор языка XAML [Электронный ресурс] – режим доступа до статті:
<http://msdn.microsoft.com/ru-ru/library/windows/apps/hh700354.aspx>

ДОДАТОК А

Титульний лист пояснювальної записки

ЗАТВЕРДЖЕНО

Наказ Міністерства освіти і науки,
молоді та спорту України
29 березня 2012 року № 384

Форма № Н-6.01

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТкафедра Програмних засобів

(повна назва кафедри, циклової комісії)

**КУРСОВИЙ ПРОЕКТ
(РОБОТА)**з дисципліни "Розробка застосувань для мобільних пристроїв"

(назва дисципліни)

на тему "Розробка мобільного додатку – гри «4 в ряд» для мобільної
операційної системи IOS"Студента (ки) 5 курсу 219 групи
напряму підготовки 6.050101
спеціальності 7.05010104С. І. Сидорова

(прізвище та ініціали)

Керівник доц., к.т.н О.О. Степаненко

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Національна шкала _____

Кількість балів: _____ Оцінка: ECTS _____

Члени комісії

(підпис)

(прізвище та ініціали)

(підпис)

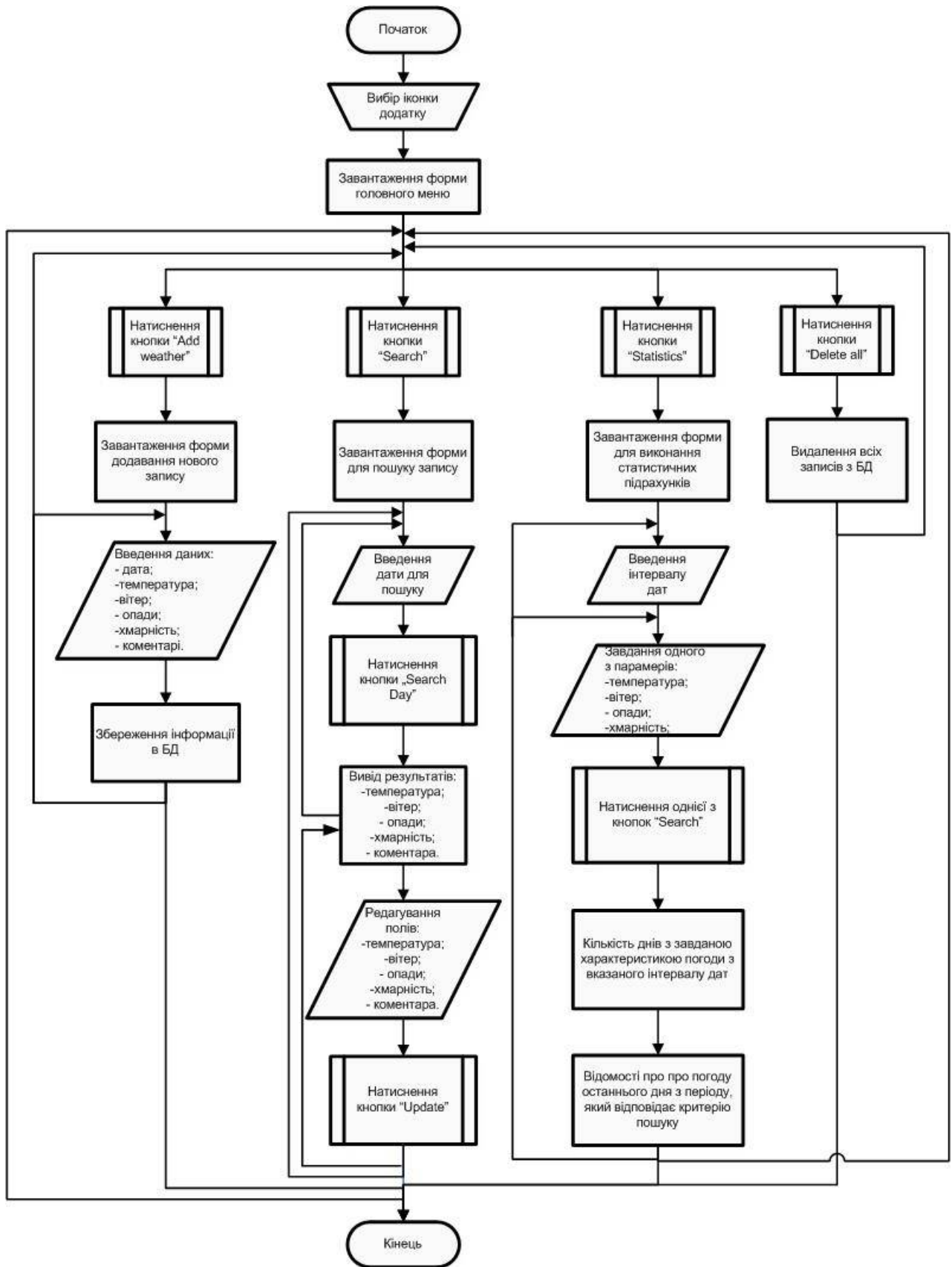
(прізвище та ініціали)

(підпис)

(прізвище та ініціали)

м.Запоріжжя
2013

ДОДАТОК Б
Блок-схема програми (приклад)



ДОДАТОК В
ТЕКСТ МОБІЛЬНОГО ДОДАТКУ

B.4 Текст файла StatisticsActivity.java

```
package com.example.weatherjournal;

import java.sql.Date;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import android.app.Activity;
import com.example.weatherjournal.AddNewActivity.*;
import android.content.Intent;
import android.os.Bundle;
import android.view.Gravity;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.text.Html;
import android.util.Log;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;
import android.widget.AdapterView.OnItemSelectedListener;
import java.text.DateFormat;
public class StatisticsActivity extends Activity implements OnClickListener{

    private final String[] base_windS = {"","strong","middle","slight", "no wind"};
    private final String[] base_precipitationS = {"","rain","snow","storm", "hail"};
    private final String[] base_cloudinessS = {"","clear","partly cloudy","cloudy",
"mainly cloudy"};

    Button SearchT, SearchW, SearchP,SearchC;
    EditText Date1, Date2, TemperatureSS1, WindSS, PrecipitationSS,
CloudinessSS,Count;
    EditText
ListDate,ListTemperature,ListWind,ListPrecipitation,ListCloudiness,ListComments;

    static final String LOG_TAG = "myLogs";

    int IDstartData;
    int IDendData;

    DBHelper dbHelper;

    public void onCreate(Bundle savedInstanceState){
        //загрузка формы statistics
        super.onCreate(savedInstanceState);
        setContentView(R.layout.statistics);

        //получение id кнопок на форме
        SearchT = (Button) findViewById(R.id.SearchT);
        SearchT.setOnClickListener(this);
```

```

SearchW = (Button) findViewById(R.id.SearchW);
SearchW.setOnClickListener(this);

SearchP = (Button) findViewById(R.id.SearchP);
SearchP.setOnClickListener(this);

SearchC = (Button) findViewById(R.id.SearchC);
SearchC.setOnClickListener(this);

//получение id полей на форме
Date1=(EditText) findViewById(R.id.Date1);
Date2=(EditText) findViewById(R.id.Date2);
TemperatureSS1 = (EditText) findViewById(R.id.TemperatureSS1);
WindSS = (EditText) findViewById(R.id.WindSS);
PrecipitationSS = (EditText) findViewById(R.id.PrecipitationSS);
CloudinessSS = (EditText) findViewById(R.id.CloudinessSS);
Count = (EditText) findViewById(R.id.Count);
ListDate = (EditText) findViewById(R.id.ListDate);
ListTemperature = (EditText) findViewById(R.id.ListTemperature);
ListWind = (EditText) findViewById(R.id.ListWind);
ListPrecipitation = (EditText) findViewById(R.id.ListPrecipitation);
ListCloudiness = (EditText) findViewById(R.id.ListCloudiness);
ListComments = (EditText) findViewById(R.id.ListComments);

// создаем объект для создания и управления версиями БД
dbHelper = new DBHelper(this);

//Выпадающие список для поля Wind_
ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_item, base_winds);

adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
WindSS = (EditText) findViewById(R.id.WindSS);
Spinner spinner = (Spinner) findViewById(R.id.spinner);
spinner.setAdapter(adapter);
spinner.setPrompt("Ветер");// заголовок
spinner.setSelection(0);// выделяем элемент
spinner.setOnItemClickListener(new OnItemClickListener() {
    public void onItemClick(AdapterView<?> parent, View view,int
position, long id){
        String ans = base_winds[position];
        WindSS.setText(ans);
    }
    public void onNothingSelected(AdapterView<?> arg0) {}
});

//Выпадающие список для поля Precipitation_
ArrayAdapter<String> adapter2 = new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_item, base_precipitationS);

adapter2.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
PrecipitationSS = (EditText) findViewById(R.id.PrecipitationSS);
Spinner spinner2 = (Spinner) findViewById(R.id.spinner2);
spinner2.setAdapter(adapter2);

```

B.16 Текст файла WeatherJornalManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.weatherjournal"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="18" />
    <application
        android:allowBackup="true"
        android:icon="@drawable/sun_"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name="com.example.weatherjournal.MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity
            android:name="com.example.weatherjournal.AddNewActivity"
            android:label="@string/app_name" >
        </activity>

        <activity
            android:name="com.example.weatherjournal.SearchActivity"
            android:label="@string/app_name" >
        </activity>

        <activity
            android:name="com.example.weatherjournal.StatisticsActivity"
            android:label="@string/app_name" >
        </activity>

    </application>
</manifest>
```

ДОДАТОК А

ТЕКСТ МОБІЛЬНОГО ДОДАТКУ

A1. Текст файлу MainPage.xaml.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Navigation;
using Microsoft.Phone.Controls;
using Microsoft.Phone.Shell;
using TicTacToe.Resources;

namespace TicTacToe
{
    public partial class MainPage : PhoneApplicationPage
    {
        public MainPage()
        {
            InitializeComponent();

            private void StartGame_Click(object sender, RoutedEventArgs e)
            {
                NavigationService.Navigate(new Uri("/Game.xaml", UriKind.Relative));
            }

            private void Rules_Click(object sender, RoutedEventArgs e)
            {
                NavigationService.Navigate(new Uri("/Rules.xaml", UriKind.Relative));
            }

            private void About_Click(object sender, RoutedEventArgs e)
            {
                NavigationService.Navigate(new Uri("/About.xaml", UriKind.Relative));
            }
        }
    }
}
```