

# БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

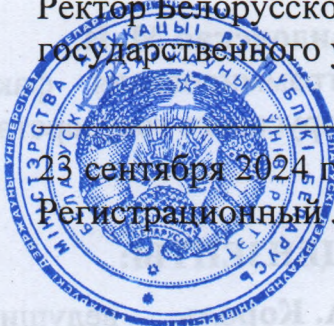
УТВЕРЖДАЮ

Ректор Белорусского  
государственного университета

А.Д.Король

23 сентября 2024 г.

Регистрационный № 2194/6.



## ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ ДЛЯ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Учебная программа учреждения образования по учебной дисциплине для  
специальности:

**6-05-0533-11 Прикладная информатика**

2024 г.

Учебная программа составлена на основе ОСВО 6-05-0533-11-2023, учебных планов БГУ: № 6-5.3-59/03, № 6-5.3-59/04, № 6-5.3-59/05 от 15.05.2023, № 6-5.3-59/11ин, № 6-5.3-59/12ин., № 6-5.3-59/13ин от 31.05.2023.

#### **СОСТАВИТЕЛИ:**

**Давидовская М. И.**, старший преподаватель кафедры технологий программирования факультета прикладной математики и информатики Белорусского государственного университета.

#### **РЕЦЕНЗЕНТЫ:**

**И. А. Король**, ведущий научный сотрудник, заместитель директора государственного предприятия «Центр систем идентификации» Национальной академии наук Беларуси, кандидат физико-математических наук, доцент

**Т. В. Соболева**, доцент кафедры многопроцессорных систем и сетей факультета прикладной математики и информатики Белорусского государственного университета, кандидат физико-математических наук

#### **РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:**

Кафедрой технологий программирования БГУ  
(протокол № 2 от 12.09.2024)

Научно-методическим советом БГУ  
(протокол № 2 от 19.09.2024)

Заведующий кафедрой  
технологий программирования \_\_\_\_\_ А. Н. Курбацкий

## **ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

### **Цели и задачи учебной дисциплины**

**Цель учебной дисциплины** — подготовка специалиста по проектированию и разработке мобильных приложений на языках Objective-C и Swift для операционной системы iOS. Формирование и развитие системы знаний по теоретическим и практическим аспектам проектирования и разработки приложений, получаемых будущими специалистами в процессе учёбы и необходимых им в дальнейшем для успешной работы по разработке мобильных приложений

### **Задачи учебной дисциплины:**

1. Изучить технологии проектирования и разработки консольных и мобильных приложений;
2. Изучить синтаксис языков программирования Objective-C и Swift;
3. Сформировать навыки проектирования и разработки мобильных приложений;
4. Сформировать навыки командной работы и управления ИТ-проектами.

**Место учебной дисциплины** в системе подготовки специалиста с высшим образованием.

Учебная дисциплина относится к модулю «Программирование» государственного компонента.

Компетенции в области проектирования и разработки мобильных приложений относятся к ключевым практическим навыкам, которыми должны владеть специалисты и выпускники вузов в области разработки программного обеспечения.

Учебно-методическая карта дисциплины сформирована таким образом, что способствует приобретению навыков управления исходным кодом проектов с помощью систем контроля версий, автоматизации разработки многофайловых проектов, созданию приложений на основе баз данных, проектированию технической документации и спецификации программного продукта, управления ИТ-проектами, формированию навыков в производственно-технологической и экспертно-консультационной деятельности.

Содержание учебной программы соответствует уровню подготовленности студентов к изучению данной дисциплины и основывается на базовых учебных дисциплинах I ступени высшего образования, включая «Основы и методология программирования», «Разработка кросс-платформенных приложений», «Промышленное программирование».

Изучение дисциплины «Технологии программирования для мобильных приложений» будет востребовано при изучении дисциплины «Программирование мобильных и встраиваемых систем» и других дисциплин модуля «Прикладное программирование», дисциплин профилизации и при выполнении курсовых и дипломных работ.

### **Требования к компетенциям**

Освоение учебной дисциплины «Технологии программирования для мобильных приложений» должно обеспечить формирование следующих

универсальных, базовых профессиональных и специализированных компетенций:

**универсальные компетенции:**

Быть способным к саморазвитию и совершенствованию в профессиональной деятельности.

Проявлять инициативу и адаптироваться к изменениям в профессиональной деятельности.

**базовые профессиональные компетенции:**

Строить, анализировать и тестировать алгоритмы и программы решения типовых задач обработки информации с использованием структурного, объектно-ориентированного подходов и иных парадигм программирования.

**специализированные компетенции:**

Использовать программные средства и технологии для создания прикладного программного обеспечения.

В результате освоения учебной дисциплины студент должен:

**знать:**

- методологии управления ИТ-проектами;
- методы проектирования архитектуры и пользовательского интерфейса программных систем;
- методы разработки мобильных приложений с использованием различных инструментальных средств.

**уметь:**

- применять систему контроля версий для управления исходным кодом проектов;
- проектировать и использовать базу данных в программных проектах;
- разрабатывать мобильные приложения на языках Objective-C и Swift.

**иметь навык:**

- проектирования спецификации и управления ИТ-проектом;
- версионирования исходного кода;
- проектирования и разработки мобильных приложений.

## **Структура учебной дисциплины**

Дисциплина изучается в 4 семестре. В соответствии с учебным планом всего на изучение учебной дисциплины «Технологии программирования для мобильных приложений» отведено для очной формы получения высшего образования — 216 часов, в том числе 70 аудиторных часов, из них: лекции — 34 часа, лабораторные занятия — 36 часов. **Из них:**

лекции — 34 часа, лабораторные занятия — 30 часов, управляемая самостоятельная работа — 6 часов.

Трудоёмкость учебной дисциплины составляет 6 зачётных единиц.

Форма промежуточной аттестации — **зачёт, экзамен.**

## СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

### Раздел 1 Технологии управления, документирования и сборки приложений

#### ***Тема 1.1 Основы разработки приложений и управления проектами в окружении ОС macOS***

Эволюция операционных систем семейства Unix и концепция macOS. Графический и командный пользовательский интерфейс macOS. Командный интерпретатор на примере `bash` и `zsh`. Базовые команды, права доступа. Справочные системы. Пакетный менеджер и способы установки приложений. Файловая система macOS. Технологии виртуализации и запуск macOS в виртуальной машине

Введение в управление ИТ-проектами. Гибкие методологии разработки программного обеспечения и их характеристика. Применение Kanban для управления проектами. Онлайн-сервисы для управления проектами.

#### ***Тема 1.2 Управление кодом и документирование проекта в продуктовой разработке***

Системы контроля версий (СКВ) для управления исходным кодом приложений и их типы. Концепция системы контроля версий. Локальный и внешний репозитории и основные команды управления. Локальные и внешние ветки. Инструментальные средства `git`.

Язык разметки Markdown. Документирование проекта с помощью файла README. Документирование проекта в wiki. Github Pages и другие сервисы документирования проектов.

#### ***Тема 1.3 Многофайловое программирование и автоматизация разработки***

Концепция разработки на основе модели «Клиент-Интерфейс-Сервер» (КИС). Многофайловое программирование и сборка проекта с помощью `make`. Файл Makefile, его структура и синтаксис. Современные решения для автоматической сборки и распространения. Онлайн-сервисы для разработки, управления и сборки проектов на примере Repl.it и других.

Технологии непрерывной интеграции: характеристики и возможности. Основные понятия, файлы конфигурации и характеристики Github Actions. Другие решения для непрерывной интеграции и их интеграция с системой контроля версий.

#### ***Тема 1.4 Проектирование приложений на основе базы данных SQLite***

Основы SQLite. Типы данных SQLite и правила определения соответствия типов. Консоль SQLite и основные команды управления базой данных (БД).

Язык структурированных запросов SQL и стандарты SQL. Категории команд SQL. Выборки и параметрические запросы.

Разработка приложений на языке C и хранение данных в БД SQLite. Объектная модель SQLite C API. Объект-соединение и методы взаимодействия

с базой данных SQLite. Создание и удаление таблиц. Вставка, выборка, обновление и удаление данных. Транзакции в SQLite.

### ***Тема 1.5 Разработка требований и спецификации ПО***

Требования к ПО и их виды. Документирование требований. Унифицированный язык моделирования (Unified Modeling Language — UML) для проектирования ПО. Классификация диаграмм UML. Диаграммы вариантов использования, деятельности, состояний, последовательности, объектов, классов, пакетов, компонентов и размещения и др., их элементы и связи между элементами. Правила составления спецификации и технического задания на разработку ПО.

## **Раздел 2 Технологии разработки мобильных приложений для ОС iOS**

### ***Тема 2.1. Введение в разработку мобильных приложений для операционной системы iOS***

Введение в операционную систему iOS. Организация пользовательского интерфейса в iOS. Архитектура iOS и слои абстракции. Типы API в macOS и iOS. Структура SDK macOS, SDK iOS и ключевые библиотеки.

Инструменты для разработки iOS-приложений. Возможности IDE Xcode. Конфигурация проекта, файл списка свойств формата .plist. Настройка симулятора. Редактирование, документирование и комментирование кода. Управление версиями и отладка в Xcode. Xcode.

### ***Тема 2.2. Базовые и расширенные средства языка программирования Objective-C***

Основы языка программирования Objective-C. Токены, идентификаторы и зарезервированные слова в Objective-C. Типы данных, квалификаторы, константы, выражения и переменные в Objective-C. Операторы Objective-C. Ввод и вывод в Objective-C. Комментирование кода в Objective-C.

Строки и управление строковым типом данных. Структуры и указатели в Objective-C. Коллекции в Objective-C и их характеристики. Массивы в Objective-C и их методы. Словари в Objective-C и их методы. Наборы в Objective-C и их методы.

Управляющие конструкции на примере операторов ветвления и циклов в Objective-C.

Понятие объекта и класса. Объявление и реализация классов. Объявление, выделение памяти, инициализация экземпляра класса и механизмы освобождения памяти. Синтезируемые методы доступа классов. Области видимости класса и экземпляра класса. Концепция наследования классов в Objective-C.

Функции в Objective-C. Объявление функций. Области действия функций и спецификатор static. Переменные в функциях.

Протоколы в Objective-C и их концепция. Методы протокола. Наследование протоколов. Поддержка протокола классом.

Структура консольных проектов на языке Objective-C. Определение интерфейсов классов, их реализации и блока программы согласно модели КИС.

### ***Тема 2.3. Базовые средства и контейнерные типы в языке Swift***

Язык программирования Swift. Переменные и константы. Ввод и вывод информации в языке Swift. Комментирование кода.

Базовые типы данных. Копирование типа. Операторы в языке Swift. Приведение базовых типов данных. Псевдонимы типов. Методы базовых типов данных. Кодировка Unicode в строковых типах данных.

Управляющие конструкции. Утверждения. Оператор условия if и оператор ветвления switch. Циклы.

Контейнерные типы данных и их характеристики. Понятие последовательности и коллекции. Кортежи: литерал кортежа, объявление кортежа, тип данных кортежа, явное и неявное указание типа, доступ к элементам кортежа и операции над кортежами. Диапазоны: полуоткрытый и закрытый диапазон, бинарная, префиксная и постфиксная формы диапазона, свойства и методы диапазонов. Массивы в Swift, их свойства и методы. Словари в Swift, их свойства и методы. Наборы в Swift, их свойства и методы.

Структура консольных проектов на языке Swift и управление зависимостями с помощью SwiftPM.

### ***Тема 2.4. Опциональные типы и расширенные средства языка Swift***

Опциональные типы данных. Извлечение опционального значения. Проверка значения опционального типа данных. Опциональное связывание.

Объявление функций, входные параметры и возвращаемое значение. Функциональный тип. Функция в качестве входного и возвращаемого значений.

Объектные типы данных: структуры, перечисления и классы. Перечисления: синтаксис, параметры, свойства и методы. Структуры, их синтаксис, свойства и методы. Классы, их синтаксис, свойства и методы. Требования к классам и типы свойств. Наследование классов.

Протоколы, их свойства и методы. Наследование протоколов.

Структура проектов на языке Swift. Определение функций и их вызов, объявление интерфейсов классов и их реализация в многофайловых проектах на языке Swift.

### ***Тема 2.5. Проектирование мультязычного пользовательского интерфейса iOS-приложения***

Проектирование интерфейса мобильного приложения в Interface Builder и библиотека UIKit. Основные элементы управления и их виды. Представления View и их типы. Привязка элементов интерфейса и обработка событий. ViewController и его жизненный цикл. AutoLayout и адаптация элементов интерфейса под разные устройства. Использование ресурсов приложения. Понятие делегирования. Обработка ввода текста. Жизненный цикл мобильного приложения.

Проектирование интерфейса мобильного приложения и библиотека SwiftUI. Ключевые понятия и элементы интерфейса SwiftUI. Управление свойствами представлений и переходами между ними.

Структура проекта мобильного приложения на языке Objective-C и Swift. Архитектурные шаблоны MVC и MVVM и их применение.

Интернационализация и локализация. Локализация интерфейса и файлы локализации. Локализация макетов storyboard. Интернационализация изображений, форматов данных, иконок приложений. Словарь конфигурации спецификатора спецификации формата (поддержка множественного числа). Локализация проектов на основе SwiftUI.

### ***Тема 2.6. Графика, анимация, обработка жестов в мобильных приложениях***

Библиотека Core Graphics и её основные возможности. Геометрические типы данных. Графические примитивы и двумерная графика. Цвета и шрифты. Рисование. Использование камеры. Селектор изображений и доступ к галерее изображений.

Анимация представлений и анимация слоёв. Возможности библиотеки Core Animation. Анимация и анимируемые свойства UIView. Анимация с помощью CALayer и его наследников.

Жесты и виды жестов. Обработка жестов. Методы уведомления о касаниях. Обработка одного и нескольких жестов. Распознавание жестов и UIGestureRecognizer.

### ***Тема 2.7. Определение местоположения и хранение данных в iOS-приложениях***

Библиотека Core Location. Определение местоположения, основные подходы и методы. Разрешение на использование служб определения местоположения. Диспетчер местоположения CLLocationManager. Требуемая точность определения местоположения и фильтры расстояний. Обновление координат и определение расстояния между объектами CLLocation.

Библиотека MapKit. Визуализация карты. Понятие маркера и MKAnnotation. MKDirections (направления) и MKOverlay («наложения»). Методы делегата MKMapViewDelegate. Автоматическая группировка маркеров при масштабировании.

Механизмы хранения данных в файловой системе iOS: списки свойств, архивы объектов, SQLite3, Core Data. Каталоги приложения (Documents, Library и tmp) и права доступа. Стратегии хранения файлов их преимущества и недостатки.

Списки свойств. Запись и считывание. Сериализация списка свойств. Управление настройками приложения и класс UserDefaults (NSUserDefaults).

SQLite3 и способы подключения библиотеки в проект. Открытие и закрытие базы данных. Создание и наполнение таблиц БД. Извлечение данных из БД. Связанные переменные.

Библиотека Core Data. Атрибуты, сущности и объекты. Свойства сущностей. Разработка модели данных. Извлечение значений из атрибута и запись в атрибут. Контекст и использование контекста. Типы хранилищ.

### ***Тема 2.8. Тестирование и публикация мобильных приложений***

Модульное тестирование. Преимущества и недостатки модульного тестирования. Программное средство XCTest и разработка модульных тестов. Тестирование пользовательского интерфейса (User Interface — UI) и разработка UI-тестов в Xcode.

Программа Apple Developer и публикация приложений в AppStore.

## УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная (дневная) форма получения высшего образования с применением дистанционных образовательных технологий  
(ДОТ)

| Номер раздела,<br>темы | Название раздела, темы                                                         | Количество аудиторных часов |                         |                        |                         |      | Количество часов<br>УСР | Форма контроля                                        |
|------------------------|--------------------------------------------------------------------------------|-----------------------------|-------------------------|------------------------|-------------------------|------|-------------------------|-------------------------------------------------------|
|                        |                                                                                | Лекции                      | Практические<br>занятия | Семинарские<br>занятия | Лабораторные<br>занятия | Иное |                         |                                                       |
| 1                      | 2                                                                              | 3                           | 4                       | 5                      | 6                       | 7    | 8                       | 9                                                     |
| <b>1</b>               | <b>Технологии управления,<br/>документирования и сборки<br/>приложений</b>     | <b>14</b>                   |                         |                        | <b>12</b>               |      | <b>4</b>                |                                                       |
| 1.1                    | Основы разработки приложений и<br>управления проектами в окружении ОС<br>macOS | 2                           |                         |                        |                         |      |                         | Опрос                                                 |
| 1.2                    | Управление кодом и документирование<br>проекта в продуктовой разработке        | 2                           |                         |                        | 2                       |      | 2                       | Отчёт по<br>лабораторной работе<br>Контрольная работа |
| 1.3                    | Многофайловое программирование и<br>автоматизация разработки                   | 2                           |                         |                        | 2                       |      | 2                       | Отчёт по<br>лабораторной работе<br>Контрольная работа |
| 1.4                    | Проектирование приложений на основе<br>базы данных SQLite                      | 4                           |                         |                        | 4                       |      |                         | Отчёт по<br>лабораторной работе<br>Контрольная работа |
| 1.5                    | Разработка требований и спецификация<br>ПО                                     | 4                           |                         |                        | 4                       |      |                         | Отчёт по<br>лабораторной работе                       |

|          |                                                                          |           |  |  |           |  |          |                                                                  |
|----------|--------------------------------------------------------------------------|-----------|--|--|-----------|--|----------|------------------------------------------------------------------|
| <b>2</b> | <b>Технологии разработки мобильных приложений для ОС iOS</b>             | <b>20</b> |  |  | <b>18</b> |  | <b>2</b> |                                                                  |
| 2.1      | Введение в разработку мобильных приложений для операционной системы iOS  | 1         |  |  |           |  |          | Опрос                                                            |
| 2.2      | Базовые и расширенные средства языка программирования Objective-C        | 3         |  |  | 2         |  |          | Отчёт по лабораторной работе<br>Контрольная работа               |
| 2.3      | Базовые средства и контейнерные типы в языке Swift                       | 2         |  |  | 2         |  |          | Отчёт по лабораторной работе<br>Контрольная работа               |
| 2.4      | Опциональные типы и расширенные средства языка Swift                     | 2         |  |  | 2         |  |          | Отчёт по лабораторной работе<br>Контрольная работа               |
| 2.5      | Проектирование мультязычного пользовательского интерфейса iOS-приложения | 4         |  |  | 2         |  | 2        | Отчёт по лабораторной работе                                     |
| 2.6      | Графика, анимация, обработка жестов в мобильных приложениях              | 2         |  |  | 4         |  |          | Отчёт по лабораторной работе<br>Контрольная работа               |
| 2.7      | Определение местоположения и хранение данных в iOS-приложениях           | 4         |  |  | 4         |  |          | Отчёт по лабораторной работе<br>Контрольная работа<br>Коллоквиум |
| 2.8      | Тестирование и публикация мобильных приложений                           | 2         |  |  | 2         |  |          | Отчёт по лабораторной работе                                     |

## ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

### Основная литература

1. Баланов, А. Н. Комплексное руководство по разработке: от мобильных приложений до веб-технологий : учебное пособие для вузов / А. Н. Баланов. — Санкт-Петербург: Лань, 2024. — 412 с. — Режим доступа: <https://e.lanbook.com/book/394577>. — Дата доступа: 16.05.2024.
2. Казанский А. А. Разработка приложений на Swift 5.1 и SwiftUI с нуля. / А. А. Казанский. — СПб.: БХВ-Петербург, 2021. — 381 с. — <https://ibooks.ru/reading.php?short=1&productid=385778>.
3. Маран, М. М. Программная инженерия : учебное пособие для вузов / М. М. Маран. — 3-е изд., стер. — Санкт-Петербург: Лань, 2022. — 194 с. — Режим доступа: <https://e.lanbook.com/book/189470>. — Дата доступа: 18.05.2024.
4. Усов, В. Swift. Основы разработки приложений под iOS, iPadOS и macOS / Василий Усов. — 5-е изд., доп. и перераб. — Санкт-Петербург [и др.] : Питер, 2020. — 495 с. — URL: <https://ibooks.ru/reading.php?short=1&productid=374467>.

### Дополнительная литература

1. Рощин, П. Г. Командная разработка программного обеспечения с помощью системы контроля версий Git: Конспект лекций: учебное пособие / П. Г. Рощин. — Москва: НИЯУ МИФИ, 2022. — 108 с. — Режим доступа: <https://e.lanbook.com/book/355550>. — Дата доступа: 16.05.2024.
2. Усов, В. Swift. Разработка приложений под iOS на основе фреймворка UIKit. / В. Усов. — Москва, 2021. — 492 с.: ил.
3. A Swift Tour. [Электронный ресурс] / Apple Inc. — 2024. — Режим доступа: <https://docs.swift.org/swift-book/GuidedTour/GuidedTour.html>. — Дата доступа: 10.04.2024.
4. Chakon Sc., Straub B. Pro Git. [Электронный ресурс] / Apress. — Second Edition — 2024. — Режим доступа: <https://git-scm.com/book/ru/v2>. — Дата доступа: 20.04.2024.
5. Objective-C 2.0 Essentials. [Электронный ресурс] / Payload Media. — Second Edition — 2016. — Режим доступа: [https://www.techotopia.com/index.php/Objective-C\\_2.0\\_Essentials](https://www.techotopia.com/index.php/Objective-C_2.0_Essentials). — Дата доступа: 14.04.2024.
6. SQL As Understood By SQLite. [Электронный ресурс] / SQLite Consortium. — 2024. — Режим доступа: <https://www.sqlite.org/lang.html>. — Дата доступа: 18.05.2024.

### Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Объектом диагностики компетенций студентов являются знания, умения, полученные ими в результате изучения учебной дисциплины. Выявление

учебных достижений студентов осуществляется с помощью мероприятий текущей и промежуточной аттестации.

Для диагностики компетенций используются следующие средства текущей аттестации: контрольная работа; тест; коллоквиум; отчет о выполнении лабораторной работы.

Формой промежуточной аттестации по дисциплине «Технологии программирования для мобильных приложений» учебным планом предусмотрены зачёт и экзамен.

Для формирования итоговой отметки по учебной дисциплине используется модульно-рейтинговая система оценки знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая система предусматривает использование весовых коэффициентов для текущей и промежуточной аттестации студентов по учебной дисциплине.

Формирование итоговой отметки в ходе проведения контрольных мероприятий текущей аттестации (примерные весовые коэффициенты, определяющие вклад текущей аттестации в отметку при прохождении промежуточной аттестации):

- опрос — 5%;
- контрольная работа — 30%;
- коллоквиум — 10%;
- отчёт по лабораторной работе — 55%.

Итоговая отметка по дисциплине рассчитывается на основе итоговой отметки текущей аттестации (модульно-рейтинговой системы оценки знаний) 40% и экзаменационной отметки 60%.

### **Примерный перечень заданий для управляемой самостоятельной работы**

#### ***Тема 1.2 Управление кодом и документирование проекта в продуктовой разработке. (2ч.)***

Язык разметки Markdown.

Документирование проекта в репозитории проекта и способы документирования.

Создать документацию проекта приложения согласно индивидуальному заданию.

(Форма контроля — отчет по лабораторной работе).

#### ***Тема 1.3 Многофайловое программирование и автоматизация разработки. (2ч.)***

Автоматизация сборки с использованием Github Actions

Разработать конфигурационный файл автоматической сборки проекта согласно индивидуальному заданию.

(Форма контроля — отчет по лабораторной работе).

#### ***Тема 2.5 Проектирование пользовательского мультязычного интерфейса iOS-приложения. (2ч.)***

Интернационализация и локализация мобильных приложений.

Создать мобильное приложение с поддержкой интернационализации и локализации.

(Форма контроля — отчет по лабораторной работе).

### **Рекомендуемая тематика коллоквиума**

1. Коллоквиум «Технологии проектирования и командной разработки мобильных приложений»

### **Примерная тематика контрольных работ**

Контрольная работа 1. Управление кодом и документирование приложений.

Контрольная работа 2. Многофайловые проекты и их сборка.

Контрольная работа 3. Язык структурированных запросов SQL и его команды.

Контрольная работа 4. Функции, классы и протоколы в Objective-C.

Контрольная работа 5. Базовые и контейнерные типы в языке Swift.

Контрольная работа 6. Опциональные типы, функции и классы в Swift.

Контрольная работа 7. Базовые классы и их методы для обработки графики, анимации и жестов.

Контрольная работа 8. Базовые классы и их методы для определения, визуализации и хранения данных местоположения.

### **Примерный перечень лабораторных работ**

1. Лабораторная работа № 1. Система контроля версий git и документирование проекта.

2. Лабораторная работа № 2. Автоматизация разработки консольных приложений.

3. Лабораторная работа № 3. Разработка приложения с хранением данных в SQLite.

4. Лабораторная работа № 4. Проектирование требований и спецификации ПО.

5. Лабораторная работа № 5. Разработка консольного приложения на языке Objective-C.

6. Лабораторная работа № 6. Базовые и контейнерные типы в языке Swift.

7. Лабораторная работа № 7. Расширенные средства языка Swift и их использование.

8. Лабораторная работа № 8. Проектирование интерфейса мобильного приложения.

9. Лабораторная работа № 9. Разработка мобильного приложения с поддержкой графики, анимации и жестов.

10. Лабораторная работа № 10. Разработка мобильного приложения с определением местоположения и хранением данных в базе данных.

11. Лабораторная работа № 11. Разработка тестов и тестирование мобильных приложений.

### **Описание инновационных подходов и методов к преподаванию учебной дисциплины**

При организации образовательного процесса используется **практико-ориентированный подход**, который предполагает освоение содержание образования через решения практических задач, приобретение навыков эффективного выполнения разных видов профессиональной деятельности, реализацию групповых студенческих проектов, использование способов оценивания, фиксирующих сформированность профессиональных компетенций.

### **Методические рекомендации по организации самостоятельной работы**

Для организации самостоятельной работы студентов по учебной дисциплине следует использовать современные информационные технологии.

Разместить в сетевом доступе на образовательном портале БГУ комплекс учебных и учебно-методических материалов: учебно-программные комплексы, учебные издания для теоретического изучения дисциплины, презентации лекций, методические указания к лабораторным занятиям, электронные версии домашних заданий, материалы текущего контроля и текущей аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательных стандартов высшего образования и учебно-программной документации, в том числе вопросы для подготовки к экзамену, задания и вопросы для самоконтроля, список рекомендуемой литературы, информационных ресурсов и др.

### **Примерный перечень вопросов к зачету**

1. Графический и командный интерфейс macOS.
2. Командный интерпретатор и базовые команды.
3. Системы управления проектами в стиле Kanban.
4. Репозиторий git и базовые команды.
5. Настройка и способы инициализации git-репозитория.
6. Управление локальными и внешними ветками и репозиториями.
7. Синтаксис Markdown.
8. Синтаксис и структура Makefile.
9. Настройка конфигурационных файлов для GitHub Actions.
10. Язык SQL и категории команд.
11. Консоль SQLite и команды управления.
12. Библиотека SQLite и SQLite API для языка программирования C.
13. Структурные диаграммы UML.
14. Поведенческие диаграммы UML.
15. Разработка сценария Use Case.

16. Разработка диаграммы классов.
17. Проектирование диаграммы последовательности.
18. Создание ER-диаграммы.
19. Разработка требований в форме User Story.
20. Базовые типы и операторы языка Objective-C.
21. Массивы и словари в Objective-C.
22. Классы и протоколы в Objective-C.
23. Структура консольного проекта на языке Objective-C согласно модели КИС.
24. Базовые типы и операторы языка Swift.
25. Кортежи и диапазоны в Swift.
26. Работа с массивами в Swift.
27. Использование словарей в Swift.
28. Структура консольного проекта на языке Swift и управление зависимостями.
29. Структура проекта мобильного приложения на основе UIKit.
30. Разработка интерфейса приложения в Interface Builder.
31. Структура проекта мобильного приложения на основе SwiftUI.
32. Разработка интерфейса приложения с применением SwiftUI.
33. Библиотека UIKit и обработка жестов.
34. Библиотека Core Graphics и базовые классы.
35. Библиотека Core Animation и базовые классы.
36. Библиотека Core Location и базовые классы.
37. Библиотека Core Data и базовые классы.
38. Разработка модульных тестов с использованием XCTest.
39. Разработка тестов пользовательского интерфейса.

### **Примерный перечень вопросов к экзамену**

1. История развития, архитектура и пользовательский интерфейс macOS.
2. Файловая система macOS, основные понятия и структура.
3. Гибкие методологии управления проектами.
4. Система контроля версий и ее виды.
5. Система контроля версий git, ее концепция и основные понятия.
6. Управление локальными и внешними ветками и репозиториями в git.
7. Конфигурация проекта по модели КИС и его сборка.
8. Библиотека SQLite, возможности, типы данных и приведение типов.
9. CLI SQLite и команды управления.
10. Язык SQL и категории команд.
11. Библиотека SQLite и функции SQLite API для языка программирования C.
12. Управление соединением с базой данных с использованием C SQLite API.
13. Унифицированный язык моделирования UML, его версии и расширения.

14. Структурные диаграммы UML, примеры и характеристика.
15. Поведенческие диаграммы UML, примеры и характеристика.
16. Диаграмма вариантов использования и шаблон сценария Use Case.
17. Виды требований ПО и форматы описания.
18. Концепция и архитектура ОС iOS.
19. SDK macOS и iOS, характеристика и примеры библиотек.
20. Базовые типы, переменные и операторы в языке программирования Objective-C.
21. Операторы ветвления и циклы в языке Objective-C.
22. Коллекции, словари и их методы в языке Objective-C.
23. Массивы и их методы в языке Objective-C.
24. Классы и протоколы в Objective-C.
25. Функции в Objective-C.
26. Структура проекта Objective-C согласно модели КИС.
27. Базовые типы, переменные и операторы в языке программирования Swift.
28. Числовые и логический типы данных, приведение типов и их методы в языке Swift.
29. Строковый тип данных, приведение типов и их методы в языке Swift.
30. Управляющие конструкции и утверждения в языке программирования Swift.
31. Контейнерные типы данных в языке Swift и их характеристика.
32. Кортежи в языке программирования Swift.
33. Диапазоны в языке программирования Swift.
34. Одномерные массивы в языке программирования Swift.
35. Многомерные массивы в языке программирования Swift.
36. Наборы в языке программирования Swift.
37. Словари в языке программирования Swift.
38. Опциональный тип данных в языке программирования Swift.
39. Функции и функциональный тип данных в языке Swift.
40. Классы и протоколы в языке программирования Swift.
41. Структура проекта на языке Swift и управление зависимостями.
42. Библиотека UIKit, ключевые элементы управления и классы.
43. Проектирование адаптивного многооконного интерфейса в Interface Builder.
44. Декларативный синтаксис проектирования интерфейса и библиотека SwiftUI.
45. Структура проекта на основе UIKit и SwiftUI.
46. Интернационализация и локализация приложений в ОС iOS.
47. Библиотека Core Graphics, типы данных и основные классы.
48. Библиотека Core Animation, ключевые классы и типы анимации.
49. Обработка жестов и касаний в приложениях iOS.
50. Списки свойств и файл конфигурации проекта iOS приложения.
51. Концепция хранения данных в мобильных приложениях в iOS.

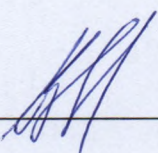
- 52. Класс UserDefaults (NSUserDefaults) и управление настройками приложения.
- 53. Библиотека SQLite в проектах на языке Swift.
- 54. Библиотека Core Location и определение местоположения.
- 55. Модель данных в Core Data, атрибуты, сущности, объекты и связи.
- 56. Ключевые классы для управления и хранения данных в Core Data.
- 57. Разработка модульных тестов и тестирование мобильных приложений.
- 58. Разработка UI-тестов в Xcode и их использование.

## ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УО

| Название учебной дисциплины, с которой требуется согласование | Название кафедры | Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине | Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола) |
|---------------------------------------------------------------|------------------|---------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| Учебная дисциплина не требует согласования                    |                  |                                                                                                               |                                                                                                   |

Заведующий кафедрой

д.т.н., профессор



А.Н.Курбацкий

12. 09 2024

# ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ ПО ИЗУ- ЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ

на \_\_\_\_ / \_\_\_\_ учебный год

| №<br>п/п | Дополнения и изменения | Основание |
|----------|------------------------|-----------|
|          |                        |           |

Учебная программа пересмотрена и одобрена на заседании кафедры  
\_\_\_\_ (протокол № \_\_\_\_ от \_\_\_\_ 202\_ г.)  
(название кафедры)

Заведующий кафедрой

\_\_\_\_\_  
(ученая степень, ученое звание)

\_\_\_\_\_  
(И.О.Фамилия)

УТВЕРЖДАЮ  
Декан факультета

\_\_\_\_\_  
(ученая степень, ученое звание)

\_\_\_\_\_  
(И.О.Фамилия)