

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
(МИНОБРНАУКИ РОССИИ)
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНБАССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ДонГТУ»)

[Handwritten signature]

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Искусственный интеллект в промышленности
(специализация)

Форма обучения	очная (очная, очно-заочная, заочная)
----------------	---

Алчевск, 2024

1 Цели и задачи изучения дисциплины

Цели дисциплины. Целью изучения дисциплины «Разработка приложений для OS iOS» создание у обучающихся компетенций в области разработки и распространений мобильных приложений для устройств, работающих под управлением операционной системы iOS.

Задачи изучения дисциплины:

- углубленное изучение мобильной операционной системы iOS;
- изучение методов и приемов алгоритмизации поставленных задач с учетом возможностей современных мобильных технологий;
- изучение основ проектирования и получение практических навыков проектирования архитектуры приложения и оценивания программных продуктов;
- получение практического опыта в выборе технологий и средств разработки ПО, а также разработки мобильного приложения с учетом целей и ограничений.

Дисциплина направлена на формирование общепрофессиональной (ОПК-8) компетенции выпускника.

2 Место дисциплины в структуре образовательной программы

Логико-структурный анализ дисциплины – курс входит в БЛОК 1 «Дисциплины (модули)», элективную часть, формируемую участниками образовательных отношений подготовки студентов по направлению 09.03.01 «Информатика и вычислительная техника».

Дисциплина реализуется кафедрой интеллектуальных систем и информационной безопасности. Основывается на базе дисциплин: «Основы программирования», «Объектно-ориентированное программирование».

Является основой для изучения следующих дисциплин: «Технологии программирования», «Подготовка к процедуре защиты и защита выпускной квалификационной работы».

Для изучения дисциплины необходимы компетенции, сформированные у студента для решения профессиональных задач деятельности, связанных с разработкой программного обеспечения.

Курс является фундаментом для ориентации студентов в сфере разработки программного обеспечения информационных систем.

Общая трудоемкость освоения дисциплины составляет 3 зачетных единицы, 108 ак.часов.

Программой дисциплины предусмотрены следующие виды занятий: лекционные (18 часов), практические (18 ч.), самостоятельная работа обучающегося составляет 72 часа.

Дисциплина изучается на 3 курсе в 6 семестре. Форма промежуточной аттестации – дифференцированный зачет.

3 Перечень результатов обучения по дисциплине, соотнесённых с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины «Разработка приложений для OS iOS» направлен на формирование компетенции, представленной в таблице 1.

Таблица 1 –Компетенции, обязательные к освоению

Содержание компетенции	Код компетенции	Код и наименование индикатора достижения компетенции
Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8	ОПК-8.3 Составляет алгоритмы, пишет и отлаживает коды на заданном языке программирования, тестирует работоспособность программ, интегрирует программные модули

4 Объём и виды занятий по дисциплине

Общая трудоёмкость учебной дисциплины составляет 2 зачётных единицы, 72 ак.ч.

Самостоятельная работа студента (СРС) включает проработку материалов лекций, подготовку к практическим занятиям, текущему контролю, выполнение индивидуального задания, самостоятельное изучение материала и подготовку к дифференцированному зачету.

При организации внеаудиторной самостоятельной работы по данной дисциплине используются формы и распределение бюджета времени на СРС для очной формы обучения в соответствии с таблицей 2.

Таблица 2 – Распределение бюджета времени на СРС

Вид учебной работы	Всего ак.ч.	Ак.ч. по семестрам
		7
Аудиторная работа, в том числе:	36	36
Лекции (Л)	18	18
Практические занятия (ПЗ)	18	18
Лабораторные работы (ЛР)	-	-
Курсовая работа/курсовой проект	-	-
Самостоятельная работа студентов (СРС), в том числе:	72	72
Подготовка к лекциям	4	4
Подготовка к лабораторным работам	9	9
Подготовка к практическим занятиям / семинарам	-	-
Выполнение курсовой работы / проекта	-	-
Расчетно-графическая работа (РГР)	-	-
Реферат (индивидуальное задание)	-	-
Домашнее задание	12	12
Подготовка к контрольным работам	-	-
Подготовка к коллоквиуму	-	-
Аналитический информационный поиск	18	18
Работа в библиотеке	18	18
Подготовка к диф.зачету	11	11
Промежуточная аттестация – диф.зачет (ДЗ)	ДЗ	ДЗ
Общая трудоемкость дисциплины		
ак.ч.	72	72
з.е.	2	2

5 Содержание дисциплины

С целью освоения компетенции, приведенной в п.3 дисциплина разбита на 4 темы:

- тема 1 (Управление потоками данных в приложениях iOS);
- тема 2 (Концепция Modal Presentation и Navigation);
- тема 3 (Ядро Core Data);
- тема 4 (Интеграция UIKit).

Виды занятий по дисциплине и распределение аудиторных часов для очной формы приведены в таблице 3.

Таблица 3 – Виды занятий по дисциплине и распределение аудиторных часов (очная форма обучения)

№ п/п	Наименование темы (раздела) дисциплины	Содержание лекционных занятий	Трудоемкость в ак.ч.	Темы практических занятий	Трудоемкость в ак.ч.	Тема лабораторных занятий	Трудоемкость в ак.ч.
1	2	3	4	5	6	7	8
1	Управление потоками данных в приложениях iOS	Использование многопоточности только в одной важной нише, связанной с HE блокировкой UI	2	–	–	Многопоточность	4
		Постоянное хранение данных с использованием UserDefaults API и JSON encoding/decoding	2	–	2	Постоянное хранение и жесты	4
		«Обертки свойств», Property Wrappers, «издатели» Publishers.	2	–	–	–	–
2	Концепция Modal Presentation и Navigation	Расширение возможностей UI с помощью методов модального представления и навигации. Получение текста от пользователя через TextField. Понимание типа KeyPath. Хранение множества EmojiArt документов.	4	–	–	Разработка интерфейса пользователя	6

Завершение таблицы 3

1	2	3	4	5	6	7	8
3	Ядро Core Data	Core Data UserDefaults. Простая. Ограниченная (только Property Lists). Очень небольшая по объему. Codable / JSON как способ преобразования практически любой структуры данных в формат JSON обмена данными и наоборот. UIDocument. Core Data Мощная. Объектно- ориентированная SwiftUI. Cloud Kit. Запоминание данных в базе данных, расположенной в «облаке». FileManager/URL/Data. Запоминание данных в Unix файловой системе в iOS.	6	—	—	Работа с данными в iOS	4
4	Интеграция UIKit	Интеграция пре-SwiftUI iOS кода в SwiftUI	2	—	—	—	—
Всего аудиторных часов			18	—		18	

6 Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации студентов по дисциплине

6.1 Критерии оценивания

В соответствии с Положением о кредитно-модульной системе организации образовательного процесса ФГБОУ ВО «ДонГТУ» (https://www.dstu.education/images/structure/license_certificate/polog_kred_modul.pdf) при оценивании сформированности компетенций по дисциплине используется 100-балльная шкала.

Перечень компетенций по дисциплине и способы оценивания знаний приведены в таблице 4.

Таблица 4 – Перечень компетенций по дисциплине и способы оценивания знаний

Код и наименование компетенции	Способ оценивания	Оценочное средство
ОПК-8	Зачет	Комплект контролирующих материалов для зачета

Всего по текущей работе в семестре студент может набрать 100 баллов, в том числе:

– защита лабораторных работ – всего 100 баллов;

Зачет проставляется автоматически, если студент набрал в течении семестра не менее 60 баллов и отчитался за каждую контрольную точку. Минимальное количество баллов по каждому из видов текущей работы составляет 60% от максимального.

Зачет по дисциплине «Разработка приложений для OS iOS» проводится по результатам работы в семестре. В случае, если полученная в семестре сумма баллов не устраивает студента, во время сессии студент имеет право повысить итоговую оценку либо в форме устного собеседования по приведенным ниже вопросам (п.п. 6.5), либо в результате тестирования.

Шкала оценивания знаний при проведении промежуточной аттестации приведена в таблице 5.

Таблица 5 –Шкала оценивания знаний

Сумма баллов за все виды учебной деятельности	Оценка по национальной шкале зачёт/экзамен
0-59	Не зачтено/неудовлетворительно
60-73	Зачтено/удовлетворительно
74-89	Зачтено/хорошо
90-100	Зачтено/отлично

6.2 Домашнее задание

Домашнее задание № 1.

Разработать собственный файл ресурсов проекта iOS, включающий картинки, текстовые файлы и настройки.

Домашнее задание № 2.

Разработать программу на iOS, которая подключается к базе данных SQLite, созданной при выполнении лабораторных работ, и выводит сведения из запроса в текстовый виджет.

6.3 Темы для рефератов (презентаций) – индивидуальное задание

Рефераты (индивидуальные задания) не предусмотрены.

6.4 Оценочные средства для самостоятельной работы и текущего контроля успеваемости

Тема 1. Управление потоками данных в приложениях iOS.

1. Что вызывает ошибку в этом фрагменте кода? Как вы могли бы ее исправить?

```
let n1: Int = 1
let n2: Float = 2.0
let n3: Double = 3.34
var result = n1 + n2 + n3
```

Ответ: В Swift неявное приведение типов между двумя типами данных невозможно. Чтобы исправить это, необходимо преобразовать каждое значение к одному типу данных. Например: `var result = Double(n1) + Double(n2) + n3`

2. Что такое Optional в Swift? Как его создать?

Ответ. Optional - это тип, который может хранить либо значение, либо nil. Вы можете создать optional, добавив вопросительный знак ? после любого типа: `var number: Int? = 10`

3. Что такое Grand Central Dispatch?

- а) Библиотека, отвечающая за последовательное выполнение кода
- б) Диспетчер, отвечающий за соединение с сетью
- в) Класс, управляющий главным потоком приложения

4. Что вызывает ошибку в этом фрагменте кода? Как вы могли бы ее

исправить?

```
let n1: Int = 1
let n2: Float = 2.0
let n3: Double = 3.34
var result = n1 + n2 + n3
```

Ответ: В Swift неявное приведение типов между двумя типами данных невозможно. `var result = Double(n1) + Double(n2) + n3`

5. Что такое `typealias` в Swift? Как его можно создать?

Ответ: `Typealias`, как следует из названия, является псевдонимом для существующего типа данных. Вы можете создать его следующим образом:

```
typealias Weight = Float
```

Тема 2. Концепция Modal Presentation и Navigation.

1. Назовите 5 утверждений передачи управления (Control Transfer Statements).

Ответ: Break, Continue, Fallthrough, Throw, Return.

2. В чем здесь проблема и как вы можете ее решить?

Рассмотрим этот фрагмент кода, который пытается получить цвет темы из локального хранилища устройства iOS:

```
var color = UserDefaults.standard.string(forKey: "themeColor")!
print(color)
```

Ответ: Чтобы исправить это, можно использовать optional binding для проверки, найдено ли значение для ключа:

```
if let color = defaults.stringForKey("themeColor") {
    print(color) }
```

3. Для чего нужны Protocols в Swift?

Ответ: протокол действует как чертеж для свойств, методов и т.д. Он описывает, как должен вести себя тип, соответствующий ему.

4. Для чего используется Defer в Swift?

Ответ: Вы можете использовать метод defer для выполнения кода перед выходом из области видимости.

5. Что такое параметр In-Out в Swift?

Ответ: параметр inout позволяет изменять значение параметра внутри функции. Чтобы сделать параметр in-out, используйте ключевое слово inout перед типом параметра. Чтобы передать переменную в качестве in-out, используйте & перед ее именем.

Тема 3. Ядро Core Data.

1. Что такое Core Data?

Ответ: Core Data — это среда для управления объектами модельного слоя в приложении на языке Swift. Она предоставляет обобщённые и автоматизированные решения распространённых задач, связанных с жизненным циклом объектов и управлением графом объектов, включая сохранение данных.

2. Пречислите основные компоненты Core Data с их кратким описанием.

Ответ:

NSManagedObject — объекты модели, предоставляемые Core Data.

NSManagedObjectModel — схема базы данных, которая описывает сущности приложения и определяет структуру управляемых объектов.

NSPersistentStoreCoordinator — связывает постоянное хранилище и модель управляемых объектов. Он позволяет сопоставить данные из хранилища, например строки базы данных SQLite, в модель объектов.

NSManagedObjectContext — управляет жизненным циклом управляемых объектов и предоставляет операции для их создания, сохранения и извлечения.

3. Определите отношение «один ко многим» в модели Core Data.

Ответ: отношение «один ко многим» в модели Core Data связывает одну сущность с несколькими экземплярами другой сущности, что полезно для представления иерархических структур данных.

4. Объясните назначение NSFetchedResultsController.

Ответ: NSFetchedResultsController управляет результатами запроса на получение основных данных для отображения в UITableView или UICollectionView. Он эффективно обрабатывает большие наборы данных и обновляет пользовательский интерфейс при изменении данных.

5. Объясните роль NSManagedObjectID.

Ответ: NSManagedObjectID однозначно идентифицирует управляемый объект в Core Data, подобно первичному ключу в базе данных. Это обеспечивает безопасное обращение к объектам в разных контекстах и эффективное извлечение, даже если объект имеет сбой.

Тема 4. Интеграция UIKit.

1. Для чего используется UIKit?

а) это среда для создания приложений под iPhone, iPod touch, и iPad.

б) UIKit определяет, как вы отображаете информацию пользователю и как вы реагируете на пользовательские действия и системные события. Это также позволяет вам работать с анимациями, текстом и картинками. Помимо игр, все что вы видите на экране iOS сделано с использованием UIKit.

в) Это новый фреймворк пользовательского интерфейса, построенный на основе фреймворка UIKit. В данном языке программирования разработчики используют для написания мобильных приложений для телефонов, умных часов и компьютеров компании Apple

2. Для чего нужен класс UITableView?

а) это один из ключевых компонентов инфраструктуры UIKit, высоко оптимизирован для отображения упорядоченного списка элементов

б) Занимается динамическим вычислением позиции и размера всех view в view иерархии, на основе constraints — правил заданных для того или иного view

в) Используется для подтверждения пользователем определенных действий

г) Поле в которое мы можем ввести подсказку внутри поля формы

3. LoadingViewControlle нужен для
- а) отображения индикатора загрузки в центре своего интерфейса
 - б) Объединения шагов, необходимых для добавления или удаления child view контроллера, в два простых метода
 - в) Получения доступа к событиям
 - г) Для управления пользовательским интерфейсом
4. Что делает метод reduce?
- а) Метод в котором происходят все операции с пользовательским интерфейсом (UI)
 - б) Метод который позволяет применить переданное в него замыкание для каждого элемента коллекции
 - в) Метод который используется, когда требуется отфильтровать элементы коллекции по определенному правилу
 - г) Метод который позволяет объединить все элементы коллекции в одно значение в соответствии с переданным замыканием
5. Что такое Grand Central Dispatch?
- а) библиотека, отвечающая за последовательное выполнение кода
 - б) диспетчер, отвечающий за соединение с сетью
 - в) класс, управляющий главным потоком приложения
 - г) фреймворк, обеспечивающий работу с потоками

6.5 Вопросы для подготовки к дифзачету

- 1) С помощью какого ПО создаются мобильные приложения?
- 2) Как создать мобильное приложение в среде eclipse?
- 3) Что такое activity?
- 4) Какова структура файлов для мобильного приложения?
- 5) Какой файл является определяющим для любого мобильного приложения, и что в нём содержится?
- 6) Как называется последняя версия ОС Android?
- 7) Какие мобильные ОС Вы знаете?
- 8) Как создать файл .apk для Вашего мобильного приложения?
- 9) Что такое переменные и типы данных?
- 10) Как используются логические переменные, условный оператор if/else?
- 11) Как объявляются и используются функции?
- 12) Какие можно использовать числовые типы? Какие Вы знаете операции с ними?
- 13) Какие Вы знаете циклы?
- 14) Что такое адреса и указатели?
- 15) Как осуществляется передача по ссылке?
- 16) Что такое структуры? Как их использовать?
- 17) Что такое куча? Как ее использовать?
- 18) Что такое объекты?
- 19) Что такое сообщения?

- 20) Что такое NSString? Для чего нужен?
- 21) Что такое NSArray? Для чего нужен?
- 22) Что такое документация разработчика?
- 23) Что такое классы? Для чего нужны? Как определить?
- 24) Что такое наследование?
- 25) Что такое объектные переменные экземпляров?
- 26) Что такое предотвращение утечки памяти?
- 27) Что такое классы коллекций?
- 28) Для чего нужны константы? Как их определить?
- 29) Как произвести запись в файл с использованием NSString и NSData?
- 30) Для чего нужен обратный вызов?
- 31) Что такое протоколы?
- 32) Что такое списки свойств?
- 33) Для чего нужен init?
- 34) Что такое свойства?
- 35) Что такое категории?
- 36) Что такое блоки?
- 37) Для чего нужны префиксы?
- 38) Что такое неизменяемые объекты?
- 39) Как освободить ссылки и зачистить состояние наблюдения?
- 40) Что такое пулы автоматического освобождения?
- 41) Что такое Swift Messages?
- 42) В чем разница между операторами == и ===?
- 43) Что такое расширения типов данных?
- 44) Как использовать предикаты в fetch запросах?
- 45) Как отслеживать изменения в хранилище Core Data?
- 46) Как зашифровать хранилище Core Data?
- 47) Как создать базовый класс в Swift?

7. Учебно-методическое и информационное обеспечение дисциплины

7.1 Рекомендуемая литература

Основная литература

1. Амелин К.С. Разработка приложений для мобильных интеллектуальных систем на платформе Intel Atom [Электронный ресурс]/ К.С. Амелин [и др.].— Электрон. текстовые данные.— Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2021.— 211 с.— Режим доступа: <https://math.spbu.ru/user/gran/mat/AtomBook32.pdf> (Дата обращения 26.08.2024).

Дополнительная литература

1. Apple Developer Publications. Бесплатная серия книг для программирования на iOS. Электронный ресурс. Режим доступа: <https://www.iphones.ru/iNotes/105708?ysclid=lz6nc926tv588661895> (Дата обращения 26.08.2024).

Учебно-методические материалы и пособия

Отсутствуют.

Перечень ресурсов информационно-телекоммуникационной сети Интернет, необходимых для освоения дисциплины

1. Сайт дистанционного обучения ДонГТИ <https://moodle.dstu.education/>
2. Научная библиотека ГОУ ВО ЛНР «ДонГТИ» <https://www.dstu.education/ru/library.php>
3. Электронно-библиотечная система ФГБОУ ВО «БГТУ им. В.Г. Шухова» <http://ntb.bstu.ru>
4. ЭБС Издательства «ЛАНЬ» <http://e.lanbook.com/>

7.2 Базы данных, электронно-библиотечные системы, информационно-справочные и поисковые системы

1. Научная библиотека ДонГТУ : официальный сайт.— Алчевск. — URL: library.dstu.education.— Текст : электронный.
2. Научно-техническая библиотека БГТУ им. Шухова : официальный сайт. — Белгород. — URL: <http://ntb.bstu.ru/jirbis2/>.— Текст : электронный.
3. Консультант студента : электронно-библиотечная система.— Москва. — URL: <http://www.studentlibrary.ru/cgi-bin/mb4x>.— Текст : электронный.
4. Университетская библиотека онлайн : электронно-библиотечная система.— URL: http://biblioclub.ru/index.php?page=main_ub_red.— Текст : электронный.
5. Сайт кафедры ИСИБ <http://scs.dstu.education>

8 Материально-техническое обеспечение дисциплины

Материально-техническая база обеспечивает проведение всех видов деятельности в процессе обучения, соответствует требованиям ФГОС ВО.

Материально-техническое обеспечение представлено в таблице 9.

Таблица 9 – Материально-техническое обеспечение

[illegible]

Лист согласования рабочей программы дисциплины

Разработал:

И.о заведующего кафедрой
интеллектуальных систем
и информационной безопасности
(должность)


(подпись)

Е.Е.Бизянов.
Ф.И.О.)

(должность)

(подпись)

Ф.И.О.)

(должность)

(подпись)

Ф.И.О.)

И.о. заведующего кафедрой


(подпись)

Е.Е.Бизянов
Ф.И.О.)

Протокол № 1 заседания кафедры СКС от 27.08.2024 г.

И.о. декана факультета


(подпись)

В.В. Дьячкова.
Ф.И.О.)

Согласовано:

Председатель методической
комиссии по направлению 09.03.01
«Искусственный интеллект
в промышленности»


(подпись)

Е.Е. Бизянов
Ф.И.О.)

Начальник учебно-методического центра


(подпись)

О.А. Коваленко
Ф.И.О.)

Лист изменений и дополнений

Номер изменения, дата внесения изменения, номер страницы для внесения изменений	
ДО ВНЕСЕНИЯ ИЗМЕНЕНИЙ:	ПОСЛЕ ВНЕСЕНИЯ ИЗМЕНЕНИЙ:
Основание:	
Подпись лица, ответственного за внесение изменений	