Международный университет МИТСО  
Кафедра информационных технологий

**Курс «Технологии программирования»**

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

”GoydaMarket”

г.Минск

2025

**СОДЕРЖАНИЕ**

[1 Общие сведения 3](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940840)

[1.1 Наименование программного продукта 3](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940841)

[1.2 Шифр темы и номер документа 3](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940842)

[1.3 Наименование предприятий разработчика и заказчика 3](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940843)

[1.4 Перечень документов, на основании которых создается проект 3](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940844)

[1.5 Плановые сроки начала и окончания работы по созданию проекта 4](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940845)

[1.6 Сведения об источниках и порядке финансирования работ 4](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940846)

[1.7 Порядок оформления и предъявления заказчику результатов работ по созданию проекта 4](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940847)

[2 Назначение и цели создания программного модуль 5](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940848)

[2.1 Назначение программного модуля 5](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940849)

[2.2 Цели создания программного модуля 5](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940850)

[3 Характеристики объекта автоматизации 6](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940851)

[3.1 Краткие сведения об объекте автоматизации 6](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940852)

[3.2 Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды 6](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940853)

[4 Требования к системе 7](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940854)

[4.1 Требования к системе в целом 7](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940855)

[4.2 Требования к функциям (задачам), выполняемым системой 7](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940856)

[4.3 Требования к видам обеспечения 8](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940857)

[5 Состав и содержание работ по созданию системы 10](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940858)

[5.1 Этапы разработки 10](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940859)

[5.2 Состав работ 10](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940860)

[6 Порядок контроля и приемки системы 11](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940861)

[7 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие 12](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940862)

[8 Требования к документированию 13](file:///C:\Users\farf\Downloads\Telegram%20Desktop\default(2).docx#_Toc192940863)

# 1 Общие сведения

## Наименование программного продукта

Полное наименование продукта: GoydaMarket

## 1.2 Шифр темы и номер документа

Шифр темы и номер договора отсутствуют по причине выполнения данного проекта в рамках учебной деятельности.

## 1.3 Наименование предприятий разработчика и заказчика

### 1.3.1 Разработчики ИП Сухаревский Сёгун

### 1.3.2 Заказчик Министерство обороны РБ

## 1.4 Перечень документов, на основании которых создается проект

Разрабатываемый документ должен соответствовать законодательным, нормативным и методическим документам Республики Беларусь в сфере информационных технологий, авторских и смежных прав, в том числе в части определения прав собственности на информацию и обеспечения контроля целостности и подлинности информации.

Настоящее техническое задание оформлено в соответствии с основными требованиями к данных документам, установленными ГОСТ 34.602-89 «Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы».

## 1.5 Плановые сроки начала и окончания работы по созданию проекта

Плановый срок начала работ по созданию проекта: 08.02.2025

Плановый срок окончания работ по созданию проекта: 08.05.2025

## 1.6 Сведения об источниках и порядке финансирования работ

Так как разработка проводится в рамках учебной программы, проект не финансируется.

## 

## 1.7 Порядок оформления и предъявления заказчику результатов работ по созданию проекта

Работы по созданию сервиса «GoydaMarket», по изготовлению и наладке отдельных средств (технических, программных, информационных) и программно-технических (программно-методических) комплексов системы предоставляются раз в две недели на территории заказчика в виде отдельных отчетов, оформленных в соответствии со стандартом организации 4.2–07– 2014.

## 2. Назначение и цели создания программного модуля

### 2.1 Назначение программного модуля

Программный модуль представляет собой андроид маркетплейс приложение, разработанное на языке Kotlin для клиентской части с интеграцией серверной части, реализованной на Node.js. Приложение предназначено для организации торговли между продавцами и покупателями, позволяя публиковать объявления, осуществлять поиск товаров, вести переговоры и совершать покупки через безопасный интерфейс.

### 2.2 Цели создания программного модуля

* Обеспечить интуитивно понятный и удобный интерфейс для пользователей при просмотре товаров, оформлении заказов и коммуникации между участниками торгового процесса.
* Реализовать возможность регистрации и аутентификации пользователей с использованием современных методов защиты данных.(GoydaCrypt)
* Организовать хранение и управление информацией о товарах, категориях, объявлениях, а также историю транзакций в локальной базе данных и на сервере.
* Обеспечить масштабируемость приложения за счет разделения клиентской части (Kotlin) и серверной части (Node.js) с использованием POST API.
* Реализовать функционал поиска и фильтрации товаров, сортировки объявлений и отображения детальной информации о продуктах.
* Обеспечить интеграцию с платежными системами для проведения безопасных транзакций.
* Организовать систему уведомлений для информирования пользователей о статусе заказов, новых объявлениях и сообщениях.

## 3. Характеристики объекта автоматизации

### 3.1 Краткие сведения об объекте автоматизации

Объект автоматизации – маркетплейс приложение, предназначенное для организации электронной коммерции. Основные компоненты системы:

* **Клиентская часть:** Андроид-приложение, разработанное на Kotlin, с адаптивным дизайном для различных разрешений экрана.
* **Серверная часть:** API-сервер, разработанный на Node.js, обеспечивающий обработку запросов, управление базой данных и взаимодействие с внешними сервисами.
* **База данных:** СУБД (JSON DB) для хранения информации о пользователях, товарах, заказах и транзакциях.
* **Интеграция с платежными системами:** Обеспечение безопасного проведения платежей с использованием современных протоколов шифрования.

### 3.2 Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды

Приложение функционирует в онлайн-режиме и требует стабильного подключения к интернету для работы с серверной частью. Оптимизировано для работы на устройствах с ОС Android (версии не ниже 8.0). Серверная часть должна обеспечивать высокую доступность и масштабируемость, выдерживая большое количество одновременных запросов.

## 4. Требования к системе

### 4.1 Требования к системе в целом

#### 4.1.1 Клиентская часть:

* Реализация регистрации, аутентификации и профиля пользователя.
* Интуитивный пользовательский интерфейс для просмотра товаров, категорий, фильтров и поиска.
* Реализация возможности добавления, редактирования и удаления объявлений (для авторизованных продавцов).
* Организация функционала чата между продавцом и покупателем с использованием push-уведомлений.

#### 4.1.2 Серверная часть:

* Реализация POST API для обработки запросов от мобильного приложения.
* Обеспечение безопасности передачи данных с использованием протоколов HTTPS.
* Организация системы управления базой данных с поддержкой CRUD операций для основных сущностей (пользователи, товары, заказы, транзакции).

#### 4.1.3 Интеграция с платежными системами:

* Реализация подключения к одним или нескольким платежным шлюзам для обработки оплат.
* Обеспечение безопасности проведения транзакций с использованием современных методов шифрования и валидации.

#### 4.1.4 Интерфейс и визуальные эффекты:

* Поддержка двух цветовых схем (светлая и темная тема) с динамическим переключением.
* Использование анимаций при переходах между экранами и загрузке данных.
* Оптимизация элементов управления для сенсорного ввода.

### 4.2 Требования к функциям (задачам), выполняемым системой

#### 4.2.1 Производительность:

* Быстрое время отклика клиентского приложения даже при большом объеме данных.
* Обеспечение стабильной работы серверной части под высокой нагрузкой.

#### 4.2.2 Безопасность:

* Шифрование персональных данных пользователей.
* Аутентификация и авторизация с использованием современных стандартов (GoydaAuth).
* Регулярное обновление и патчинг серверных компонентов.

#### 4.2.3 Масштабируемость:

* Модульная архитектура, позволяющая расширять функциональность приложения.
* Возможность горизонтального масштабирования серверной части.

#### 4.2.4 Юзабилити:

* Простой и интуитивный интерфейс, адаптированный под мобильные устройства.
* Поддержка жестов и сенсорных элементов для удобной навигации.
* Локализация интерфейса на несколько языков (минимум русский и английский).

### 4.3 Требования к видам обеспечения

#### 4.3.1 Требования к математическому обеспечению системы

* Данный раздел не предъявляет специфических требований к математическому обеспечению.

#### 4.3.2 Требования к информационному обеспечению системы

* Определение структуры базы данных, модели данных и схемы связей между сущностями.
* Разработка API для взаимодействия мобильного приложения с сервером, включая документирование каждого метода.

#### 4.3.3 Требования к лингвистическому обеспечению системы

* Использование широко распространенных языков программирования: Kotlin для клиентской части, JavaScript (Node.js) для серверной.
* Документация, комментарии в коде и интерфейс приложения должны быть представлены на русском языке с возможностью переключения на английский.

#### 4.3.4 Требования к программному обеспечению системы

При разработке системы должны использоваться следующие инструменты и технологии:

**Таблица 1 – Используемые инструменты и технологии**

| **№** | **Назначение** | **Инструмент** |
| --- | --- | --- |
| 1 | Язык программирования (клиентская часть) | Kotlin |
| 2 | Язык программирования (серверная часть) | JavaScript (Node.js) |
| 3 | Интегрированная среда разработки (IDE) | Android Studio |
| 4 | СУБД | JSON |
|  |  |  |
|  |  |  |

## 5. Состав и содержание работ по созданию системы

### 5.1 Этапы разработки

#### 5.1.1 Анализ требований и проектирование:

* Анализ функциональных и нефункциональных требований.
* Определение архитектуры приложения (клиент-серверная архитектура, REST API).
* Проектирование структуры базы данных и схем взаимодействия между модулями.
* Разработка прототипа пользовательского интерфейса.

#### 5.1.2 Программирование:

* Реализация клиентской части на Kotlin:
  + Модули регистрации/аутентификации, поиска товаров, просмотра каталога и профиля пользователя.
  + Организация работы с POST API и реализация асинхронных запросов.
  + Внедрение анимационных эффектов и адаптивного интерфейса.
* Разработка серверной части на Node.js:
  + Реализация POST API для управления пользователями, товарами, заказами и транзакциями.
  + Организация системы авторизации и аутентификации (JWT, OAuth2).
  + Интеграция с выбранной СУБД.
  + Подключение платежных систем и обработка платежных транзакций.

#### 5.1.3 Тестирование:

* Проведение модульного тестирования для всех ключевых компонентов.
* Проведение интеграционных тестов клиент-серверного взаимодействия.
* Ручное тестирование пользовательского интерфейса и функциональности.
* Тестирование производительности под нагрузкой и стресс-тестирование серверной части.

#### 5.1.4 Подготовка документации:

* Создание технической документации по архитектуре, структуре кода и API.
* Разработка руководства пользователя с описанием функционала приложения, настроек и рекомендаций по эксплуатации.
* Составление отчета по результатам тестирования с перечнем выявленных ошибок и способов их устранения.

#### 5.1.5 Внедрение и сопровождение:

* Развертывание демо-версии приложения на тестовой площадке заказчика.
* Проведение обучающих сессий и инструктажей для конечных пользователей.
* Сбор обратной связи, оперативное устранение выявленных дефектов и выпуск обновлений.

### 5.2 Состав работ

* **Разработка клиентской части:**
  + Проектирование и реализация пользовательского интерфейса на Kotlin.
  + Интеграция с серверной частью посредством REST API.
  + Реализация модулей поиска, фильтрации, отображения каталога товаров и работы с чатами/уведомлениями.
* **Разработка серверной части:**
  + Проектирование архитектуры API с использованием Node.js и Express.js.
  + Создание системы управления базой данных и обеспечения безопасности данных.
  + Интеграция с платежными системами и обработка транзакций.
* **Тестирование и контроль качества:**
  + Написание и проведение автоматизированных и ручных тестов.
  + Анализ производительности и корректировки при необходимости.
* **Подготовка документации и внедрение:**
  + Техническая и пользовательская документация.
  + Организация обучающих семинаров и технической поддержки.

## 6. Порядок контроля и приемки системы

### 6.1 Контроль качества:

* Проведение модульного и интеграционного тестирования с целью проверки работоспособности каждого компонента.
* Организация тестирования интерфейса с привлечением конечных пользователей.
* Сравнение реализованного функционала с изложенными в данном ТЗ требованиями.

### 6.2 Приемка системы:

* Демонстрация работы основных функциональных блоков: регистрация/аутентификация, просмотр каталога, поиск, оформление заказа, оплата и система уведомлений.
* Проведение приемочных испытаний на стороне заказчика.
* Подписание акта приемки после устранения выявленных замечаний.

## 7. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

### 7.1 Организация обучения:

* Проведение инструктажа для пользователей, включающего обучение работе с приложением, оформлению заказов и взаимодействию с продавцами.
* Организация семинаров для технического персонала по сопровождению серверной части и администрированию базы данных.

### 7.2 Подготовка документации:

* Составление подробных руководств для пользователей (как конечных, так и администраторов).
* Разработка технической документации, описывающей архитектуру, API, процедуры резервного копирования и восстановления данных.

### 7.3 Обеспечение поддержки:

* Организация службы технической поддержки с возможностью оперативного решения проблем пользователей.
* Планирование обновлений и доработок приложения на основании отзывов пользователей и анализа работы системы.

## 8. Требования к документированию

### 8.1 Техническая документация:

* Подробное описание архитектуры системы, схем взаимодействия компонентов, структуры базы данных и алгоритмов работы серверной части.
* Документация по API с указанием всех методов, параметров и форматов данных.

### 8.2 Пользовательская документация:

* Руководство пользователя, содержащее описание интерфейса, инструкции по регистрации, поиску товаров, оформлению заказов и оплате.
* Описание функционала мобильного приложения, настройка уведомлений и работы с личным кабинетом.

### 8.3 Документация по тестированию:

* Описание методик модульного, интеграционного и нагрузочного тестирования.
* Отчеты о проведенных тестах, выявленных ошибках и способах их устранения.

# 5 Состав и содержание работ по созданию системы

## Этапы разработки

5.1.1 Анализ требований и проектирование:

— Определение архитектуры приложения и структуры данных (модели Message, ChatSession, TemplateModel).

— Разработка прототипа пользовательского интерфейса.

5.1.2 Программирование:

— Реализация основных модулей: отправка сообщений, получение ответа ИИ, управление сессиями.

— Интеграция с внешними API

— Реализация анимаций и визуальных эффектов.

* + 1. Тестирование:

— Юнит-тестирование ключевых модулей и компонентов.

— Ручное тестирование интерфейса и функциональности.

— Проверка стабильности работы при смене тем оформления.

5.1.4 Подготовка документации:

— Техническая документация по архитектуре и коду.

— Руководство пользователя с описанием функционала и инструкций по эксплуатации.

* + 1. Внедрение и сопровождение:

— Развертывание демо-версии приложения для заказчика.

— Сбор обратной связи и устранение выявленных недочётов.

## Состав работ

— Разработка клиентского приложения на JETPACKс использованием современных библиотек (http, shared\_preferences).

— Интеграция с API для генерации ответов с поддержкой потоковой передачи данных.

— Реализация системы хранения и управления чат-сессиями.

— Разработка функционала настройки и ввода API ключей.

— Организация и оформление анимационных эффектов для визуального отображения «раздумий» ИИ.

# Порядок контроля и приемки системы

6.1.1 Контроль качества**:**

— Проведение модульного тестирования и интеграционных тестов.

— Ручное тестирование пользовательского интерфейса.

— Сравнение реализованного функционала с техническими требованиями, изложенными в данном ТЗ.

* + 1. Приемка системы:

— Демонстрация всех основных функциональных блоков (отправка сообщений, получение ответа ИИ, переключение тем, настройка API).

— Подписание акта приёмки после устранения выявленных дефектов.

# 7 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

* + 1. Организация обучения:  
       Проведение инструктажа для пользователей по работе с приложением, настройке API и использованию функционала чата.
    2. Подготовка документации:  
       Предоставление подробных руководств (пользовательского и технического), описывающих настройку, эксплуатацию и обслуживание системы.
    3. Обеспечение поддержки:  
       Организация процесса технической поддержки, сбор обратной связи и планирование обновлений приложения.

# Требования к документированию

* + 1. —Техническая документация:  
       Описание архитектуры приложения, структуры кода, алгоритмов взаимодействия с API, схемы базы данных и методов хранения данных.
    2. Пользовательская документация:  
       Руководство пользователя с описанием интерфейса, инструкциями по работе с чат-приложением, настройке темы, управлению чат-сессиями и вводу API ключей.
    3. Документация по тестированию:  
       Описание методик тестирования, результаты модульного и интеграционного тестирования, перечень исправленных ошибок.

Вся работа по проектированию УН АСУ ИКИТ должна быть документирована в соответствии со стандартами. Перечень стандартов и базовых нормативных документов для выполнения проекта приведен ниже.

1. ГОСТ 34.601-90 Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания
2. ГОСТ 34.602 Информационные технологии. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.
3. РД 50-34.698-90 Автоматизированные системы требования к содержанию документов.
4. ГОСТ Р ИСО/МЭК 12207-99 Процессы жизненного цикла ПС.
5. ISO15504:1-9:1998 Оценка (аттестация) процессов жизненного цикла программных средств
6. ISO15271:1998. (ГОСТ Р-2002). ИТ. Руководство по применению ISO 12207.
7. ISO16326:1999. (ГОСТ Р-2002). ИТ. Руководство по применению ISO 12207 при административном управлении проектами.
8. ISO9000-3:1997. Стандарты в области административного управления качеством и обеспечения качества. Часть 3. Руководящие положения по применению стандарта ISO 9001 при разработке, поставке и обслуживании программного обеспечения.
9. ГОСТ 19.201-78 Единая система программной документации. Техническое здание. Требование к содержанию и оформлению.
10. ГОСТ 19.402-78 Единая система программной документации. Описание программы.
11. ГОСТ 19.404-79 Единая система программной документации. Пояснительная записка. Требования к содержанию и оформлению.
12. ГОСТ 19.301-79 Единая система программной документации. Программа и методика испытаний. Требования к содержанию и оформлению.