



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИТ)

Кафедра инструментального и прикладного программного обеспечения (ИиППО)

ОТЧЁТ ПО ПРАКТИЧЕСКИМ РАБОТАМ

по дисциплине «Проектирование информационных систем»

на тему

«Информационная система Умный дневник студента»

Выполнил студент группы ИКБО-50-23

Враженко Д.О.

Принял
Старший преподаватель

Ткаченко Д.И.

Практические работы выполнены

«__»_____2026 г.

(подпись студента)

«Зачтено»

«__»_____2026 г.

(подпись руководителя)

Москва 2026

Содержание

1. Общие сведения.....	5
1.1. Полное наименование АС и ее условное обозначение.....	5
1.2. Номер договора.....	5
1.3. Наименование организаций – Заказчика и Разработчика.....	5
1.4. Основания для разработки Подсистемы.....	5
1.5. Плановые сроки начала и окончания работ по созданию Подсистемы.....	5
1.6. Источники и порядок финансирования работ.....	5
1.7. Порядок оформления и предъявления Заказчику результатов работ.....	6
1.8. Перечень нормативно-технических документов, использованных при разработке ТЗ	6
1.9. Определения, обозначения и сокращения.....	6
1.10. Описание бизнес-ролей.....	7
2. Цели и назначение создания Подсистемы.....	8
2.1. Назначение Подсистемы.....	8
2.2. Цели создания Подсистемы.....	8
2.2.1. Повышение информированности студентов о предстоящих занятиях.....	8
2.2.2. Сокращение трудозатрат на ручное информирование.....	8
2.2.3. Уменьшение числа опозданий и пропусков занятий, связанных с незнанием расписания.....	8
3. Характеристика объектов автоматизации.....	9
3.1. Основные сведения об объекте автоматизации.....	9
3.2. Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды.....	9
4. Требования к Подсистеме.....	10
4.1. Требования к структуре Подсистемы в целом.....	10
4.1.1. Состав Подсистемы.....	10
4.1.2. Требования к способам и средствам информационного взаимодействия компонентов Подсистемы.....	10
4.1.3. Требования к режимам функционирования Подсистемы.....	10
4.1.4. Требования по диагностированию Подсистемы.....	11
4.1.5. Перспективы развития, модернизации Подсистемы.....	11
4.2. Требования к функциям (задачам), выполняемым Подсистемой.....	11
4.2.1. Сбор и подготовка данных о занятиях.....	11
4.2.2. Формирование списка получателей.....	12

4.2.3. Отправка уведомлений.....	12
4.2.4. Управление подписками.....	12
4.2.5. Логирование и отчетность.....	13
4.3. Требования к видам обеспечения Подсистемы.....	13
4.3.1. Информационное обеспечение.....	13
4.3.2. Программное обеспечение.....	14
4.3.3. Техническое обеспечение.....	14
4.3.4. Лингвистическое обеспечение.....	15
4.3.5. Организационное обеспечение.....	15
4.3.6. Методическое обеспечение.....	15
4.3.7. Математическое обеспечение.....	15
4.3.8. Метрологическое обеспечение.....	15
4.4. Общие технические требования к Подсистеме.....	16
4.4.1. Требования к численности и квалификации персонала и пользователей Подсистемы.....	16
4.4.2. Требования к показателям назначения.....	16
4.4.3. Требования к надёжности.....	16
4.4.4. Требования по безопасности.....	16
4.4.5. Требования к эргономике и технической эстетике.....	17
4.4.6. Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов Подсистемы.....	17
4.4.7. Требования к защите информации от несанкционированного доступа.....	17
4.4.8. Требования по сохранности информации при авариях.....	18
4.4.9. Требования к защите от влияния внешних воздействий.....	18
4.4.10. Требования к патентной чистоте и патентоспособности.....	18
4.4.11. Требования по стандартизации и унификации.....	19
4.4.12. Дополнительные требования.....	19
5. Состав и содержание работ по созданию Подсистемы.....	20
6. Порядок разработки Подсистемы.....	22
6.1. Порядок организации разработки Подсистемы.....	22
6.2. Перечень документов и исходных данных при разработке Подсистемы.....	22
6.3. Перечень документов, предъявляемых по окончании соответствующих этапов работ.....	23
6.4. Порядок проведения экспертизы технической документации.....	23

6.5. Перечень макетов, порядок их разработки, изготовления, испытаний, необходимость разработки на них документации, программы и методик испытаний....	23
6.6. Порядок разработки, согласования и утверждения плана совместных работ по разработке Подсистемы.....	23
6.7. Порядок разработки, согласования и утверждения программы работ по стандартизации.....	24
6.8. Требования к гарантийным обязательствам Разработчика.....	24
6.9. Порядок проведения технико-экономической оценки разработки Подсистемы.....	24
6.10. Порядок разработки, согласования и утверждения программы метрологического обеспечения, программы обеспечения надежности, программы эргономического обеспечения.....	24
7. Порядок контроля и приемки Подсистемы.....	25
7.1. Виды, состав и методы испытаний Подсистемы и ее составных частей.....	25
7.1.1. Модульное тестирование.....	25
7.1.2. Итерационное тестирование.....	25
7.1.3. Функциональное тестирование.....	25
7.1.4. Нагрузочное тестирование.....	25
7.1.5. Опытная эксплуатация.....	26
7.2. Общие требования к приемке работ, порядок согласования и утверждения приемочной документации.....	26
7.3. Статус приемочной комиссии.....	27
8. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу Подсистемы в действие.....	28
8.1. Организационные мероприятия.....	28
8.2. Технические мероприятия.....	28
8.3. Информационные мероприятия.....	28
8.4. Кадровые мероприятия и обучение персонала.....	29
8.5. Создание условий функционирования.....	29
9. Требования к документированию.....	30
9.1.1. Перечень подлежащих разработке документов.....	30
9.2. Вид представления и количество документов.....	30
9.3. Требования по использованию стандартов при разработке документов.....	31
10. Источники разработки.....	32

1. Общие сведения

1.1. Полное наименование АС и ее условное обозначение

Полное наименование: «Подсистема автоматического напоминания о занятиях» в составе информационной системы «Умный дневник студента».

Условное обозначение: «Помни УДС» (далее – Подсистема).

1.2. Номер договора

Шифр темы: ПИС-12УДС-2026.

1.3. Наименование организаций – Заказчика и Разработчика

Наименование организации-заказчика: Федеральное государственное бюджетное образовательное учреждение высшего образования «РТУ МИРЭА», кафедра инструментального и прикладного программного обеспечения (далее – Заказчик).

Наименование организации-разработчика: Враженко Даниил Олегович, студент группы ИКБО-50-23 (далее – Разработчик).

1.4. Основания для разработки Подсистемы

Работа выполняется на основании задания по дисциплине «Проектирование информационных систем».

1.5. Плановые сроки начала и окончания работ по созданию Подсистемы

Плановый срок начала работ по созданию Подсистемы – 1 марта 2026 года.

Плановый срок окончания работ по созданию Подсистемы – 30 апреля 2026 года.

1.6. Источники и порядок финансирования работ

Работы выполняются в рамках учебного процесса без дополнительного финансирования.

1.7. Порядок оформления и предъявления Заказчику результатов работ

Результаты работы представляются в виде технического задания в электронном виде (формат PDF).

1.8. Перечень нормативно-технических документов, использованных при разработке ТЗ

- ГОСТ 34.602–2020 «Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы»;
- ГОСТ 7.32–2017 «Отчёт о научно-исследовательской работе. Структура и правила оформления»;
- ГОСТ 19.701–90 «Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения».

1.9. Определения, обозначения и сокращения

Подсистема – часть информационной системы «Умный дневник студента», реализующая функцию автоматического напоминания о занятиях.

Уведомление – информационное сообщение, отправляемое студенту через выбранный канал связи (мобильное приложение, электронная почта, мессенджер).

Расписание – структурированные данные о времени, месте и дисциплинах учебных занятий.

АС – автоматизированная система.

БД – база данных.

ИС – информационная система.

ТЗ – техническое задание.

УДС – «Умный дневник студента».

API – интерфейс программирования приложений.

FCM – Firebase Cloud Messaging (сервис push-уведомлений).

1.10. Описание бизнес-ролей

Студент – пользователь Подсистемы, получающий уведомления о занятиях; имеет возможность управлять подпиской (включать/отключать, выбирать каналы и время упреждения).

Администратор – сотрудник кафедры или ИТ-отдела, отвечающий за настройку Подсистемы, мониторинг её работы и устранение сбоев.

Преподаватель (опционально) – может получать уведомления о расписании своих занятий (при необходимости).

2. Цели и назначение создания Подсистемы

2.1. Назначение Подсистемы

Подсистема предназначена для автоматизации процесса оповещения студентов о предстоящих занятиях (дисциплина, время начала, аудитория, преподаватель) путём отправки персонализированных уведомлений через доступные каналы связи (мобильное приложение, электронная почта, мессенджеры) в соответствии с настройками, выбранными каждым студентом. Подсистема используется для всех учебных групп, по которым ведётся расписание в ИС «Умный дневник студента».

2.2. Цели создания Подсистемы

2.2.1. Повышение информированности студентов о предстоящих занятиях

Целевой показатель: не менее 80% студентов, подписавшихся на уведомления, подтверждают, что получают информацию своевременно (оценка на основе периодических опросов).

2.2.2. Сокращение трудозатрат на ручное информирование

Целевой показатель: снижение количества обращений к старостам или одногруппникам с вопросами о расписании не менее чем на 50% (по сравнению с периодом до внедрения Подсистемы).

2.2.3. Уменьшение числа опозданий и пропусков занятий, связанных с незнанием расписания

Целевой показатель: снижение доли таких пропусков на 30% (по данным журнала посещаемости).

3. Характеристика объектов автоматизации

3.1. Основные сведения об объекте автоматизации

Объектом автоматизации является процесс информирования студентов о расписании учебных занятий в вузе.

В настоящее время информирование студентов осуществляется путем публикации расписания на официальном сайте вуза.

Основные недостатки существующей системы:

- отсутствие гарантии, что каждый студент своевременно ознакомится с изменениями в расписании;
- невозможность индивидуальной настройки уведомлений.

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

ИС «Умный дневник студента» функционирует круглосуточно, доступ к ней осуществляется через веб-интерфейс и мобильное приложение.

Студенты используют персональные компьютеры, смартфоны с постоянным или периодическим доступом в сеть Интернет.

Внешняя среда не оказывает влияния на функционирование программно-аппаратных средств.

4. Требования к Подсистеме

4.1. Требования к структуре Подсистемы в целом

4.1.1. Состав Подсистемы

Подсистема является функциональной частью ИС УДС и включает следующие компоненты:

- модуль сбора и анализа расписания;
- модуль управления подписками;
- модуль отправки уведомлений;
- модуль логирования и отчётности.

Все модули функционируют как единое целое, взаимодействуя через общую базу данных ИС УДС и её программный интерфейс (API).

4.1.2. Требования к способам и средствам информационного взаимодействия компонентов Подсистемы

Взаимодействие между модулями Подсистемы осуществляется через вызовы внутренних процедур и обращение к общей базе данных.

Взаимодействие с внешними сервисами (Firebase Cloud Messaging, SMTP-сервер, Telegram Bot API) производится по стандартным протоколам (HTTPS, SMTP) с использованием соответствующих библиотек.

Подсистема должна использовать существующее REST API ИС УДС для получения данных о расписании, студентах, группах и преподавателях. Формат обмена данными – JSON.

В случае недоступности API ИС УДС Подсистема должна выполнять повторные попытки подключения (не менее 3 попыток с интервалом 1 минута) и уведомлять администратора о сбое.

4.1.3. Требования к режимам функционирования Подсистемы

Основной режим – автоматическое выполнение заданий по расписанию: ежедневный сбор расписания на следующий день, отправка уведомлений за установленное время до начала занятий.

Режим настройки – администратор может изменять параметры Подсистемы: время упреждения уведомлений (по умолчанию 30 минут),

перечень доступных каналов уведомлений (push, email, Telegram), список групп, для которых выполняется рассылка, а также возможность временного отключения рассылки (например, в праздничные дни).

Режим восстановления после сбоев – при возникновении ошибок (например, сбой в работе API или внешнего сервиса) Подсистема должна автоматически перезапускать соответствующие модули и восстанавливать свою работу без потери данных. Состояние очередей уведомлений должно сохраняться в базе данных и восстанавливаться после перезапуска.

4.1.4. Требования по диагностированию Подсистемы

Подсистема должна вести журнал событий (логи) с указанием времени, типа события (информация, предупреждение, ошибка), модуля и статуса выполнения.

Логи должны храниться не менее 30 дней и быть доступны для просмотра и фильтрации администратору через веб-интерфейс или путём прямого доступа к файлам.

При возникновении критических ошибок (например, недоступность базы данных или сервиса уведомлений) должно формироваться уведомление администратору по электронной почте или через мессенджер (Telegram).

4.1.5. Перспективы развития, модернизации Подсистемы

Архитектура Подсистемы должна допускать подключение новых каналов уведомлений (SMS, VK) без изменения основной логики.

В дальнейшем возможно расширение функционала: отправка напоминаний о сдаче домашних заданий, опросов после занятий и т. п.

4.2. Требования к функциям (задачам), выполняемым Подсистемой

4.2.1. Сбор и подготовка данных о занятиях

Ежедневно в заданное время (по умолчанию в 20:00) модуль сбора запрашивает из ИС УДС расписание на следующий день.

Полученные данные структурируются по группам, дисциплинам, времени начала, аудиториям и преподавателям.

При отсутствии данных о какой-либо группе или дисциплине формируется предупреждение в лог.

4.2.2. Формирование списка получателей

Для каждой группы, имеющей занятия на следующий день, из базы данных извлекаются студенты, у которых включена подписка на напоминания.

Учитываются индивидуальные настройки подписки каждого студента: выбранные каналы уведомлений (push, email, Telegram) и время предупреждения (15, 30, 60 минут).

Студенты, отключившие подписку, исключаются из списка рассылки.

4.2.3. Отправка уведомлений

За установленное время до начала каждого занятия (с учётом индивидуальных настроек студента) модуль отправки формирует и направляет уведомления.

Содержание уведомления должно включать:

- Название дисциплины;
- Время начала в формате ЧЧ:ММ;
- Номер аудитории (если указано в расписании);
- ФИО преподавателя (если доступно).

Если аудитория или преподаватель не указаны, соответствующие поля опускаются.

Отправка уведомления считается успешной, если сервис доставки подтвердил приём сообщения (для push – подтверждение FCM, для email – успешная передача SMTP-серверу, для Telegram – успешный вызов API). При ошибке выполняются повторные попытки (до 3 раз с интервалом 1 минута). После исчерпания попытки фиксируется ошибка в лог.

4.2.4. Управление подписками

Студент может включить или отключить напоминания в своём профиле в ИС УДС.

Доступен выбор каналов уведомлений (мобильное приложение / email / Telegram) с возможностью комбинирования.

Возможность выбора времени упреждения из предопределённого списка: 15, 30, 60 минут до занятия.

Изменения в подписках должны применяться незамедлительно (в течение нескольких секунд) после сохранения в профиле.

4.2.5. Логирование и отчетность

Все отправленные уведомления записываются в лог с указанием времени отправки, адресата (ID студента), канала, содержания уведомления и статуса доставки (успешно/ошибка).

Администратор может формировать отчёты за выбранный период: общее количество отправленных уведомлений, количество успешных и ошибочных, статистика по каналам и группам.

Отчёты должны быть доступны для выгрузки в форматах CSV и PDF.

4.3. Требования к видам обеспечения Подсистемы

4.3.1. Информационное обеспечение

Подсистема использует существующие таблицы базы данных ИС УДС:

- students (идентификатор студента, ФИО, группа, email, тэг для Telegram);
- groups (название группы);
- schedule (расписание с полями: дата, время начала, время окончания, дисциплина, аудитория, преподаватель, группа);
- notification_subscriptions – новая таблица для хранения настроек подписок (создаётся в рамках разработки Подсистемы). Поля: student_id, push_enabled, email_enabled, telegram_enabled, lead_time (в минутах).

Форматы данных – реляционные таблицы СУБД (PostgreSQL/MySQL). Доступ к данным осуществляется через SQL-запросы или API.

Требования к контролю, хранению, обновлению и восстановлению данных – в соответствии с политикой резервного копирования ИС УДС (ежедневное резервирование, хранение не менее 7 дней).

4.3.2. Программное обеспечение

Язык разработки модулей – Python 3.9+ (или Java, в зависимости от стека существующей ИС УДС). Выбор языка должен быть согласован с Заказчиком.

Используемые библиотеки и фреймворки:

- Для работы с API – requests (Python) или аналоги;
- Для работы с БД – SQLAlchemy (Python) или Hibernate (Java);
- Для планирования задач – Celery (Python) с брокером Redis/RabbitMQ или встроенные средства (cron);
- Для отправки email – smtplib (Python) или JavaMail;
- Для push-уведомлений – Firebase Admin SDK;
- Для Telegram – python-telegram-bot (Python) или Telegram Bot API (Java);
- Для логирования – встроенные средства языка (log4j, logging) с ротацией файлов.

Код должен быть документирован (docstring, комментарии), структурирован по модулям, следовать стандартам оформления (PEP 8 для Python).

Разработанное ПО должно быть размещено в системе контроля версий (Git).

4.3.3. Техническое обеспечение

Подсистема развёртывается на сервере, где функционирует ИС УДС, либо на отдельном сервере, имеющем доступ к её базе данных и API.

Минимальные требования к серверу:

- Процессор: 2 ядра, 2.5 GHz;
- Оперативная память: 4 GB;
- Дисковое пространство: 20 GB (для логов и временных файлов);

- Сетевое подключение: не менее 100 Мбит/с, доступ в интернет для внешних сервисов (FCM, SMTP, Telegram API).

Для обеспечения отказоустойчивости рекомендуется использовать кластеризацию (на усмотрение администратора).

4.3.4. Лингвистическое обеспечение

Все пользовательские интерфейсы должны быть на русском языке.

Коды программ, комментарии, техническая документация – на русском или английском (по согласованию с Заказчиком).

При отправке уведомлений текст должен быть на русском языке, с возможностью локализации на английский (опционально).

4.3.5. Организационное обеспечение

Назначается администратор Подсистемы (из числа сотрудников кафедры/ИТ-отдела), ответственный за первоначальную настройку, мониторинг и устранение сбоев.

Студенты самостоятельно управляют подписками через интерфейс ИС УДС, дополнительное обучение не требуется.

При возникновении нештатных ситуаций администратор действует в соответствии с инструкцией (разрабатывается в рамках документации).

4.3.6. Методическое обеспечение

Разрабатываются следующие документы:

- Руководство администратора по настройке и эксплуатации Подсистемы;
- Инструкция для студентов по управлению подписками.

Документы оформляются в электронном виде (PDF) и публикуются в разделе «Помощь» ИС УДС или передаются Заказчику отдельно.

4.3.7. Математическое обеспечение

Требования не предъявляются.

4.3.8. Метрологическое обеспечение

Требования не предъявляются.

4.4. Общие технические требования к Подсистеме

4.4.1. Требования к численности и квалификации персонала и пользователей Подсистемы

Администратор – 1 человек, владеющий навыками администрирования Linux/Windows, базовыми знаниями SQL и работы с API, пониманием принципов работы push-уведомлений и email-рассылок.

Пользователи – все студенты вуза. Дополнительного обучения не требуется, достаточно краткой инструкции.

4.4.2. Требования к показателям назначения

Время реакции на изменение расписания: изменения, внесённые в ИС УДС до 18:00 текущего дня, должны учитываться при рассылке на следующий день.

Время отправки уведомления одному студенту – не более 1 секунды (среднее).

Пропускная способность: возможность отправки не менее 100 уведомлений в секунду.

Доступность Подсистемы: не менее 99% времени в месяц (за исключением плановых профилактических работ).

4.4.3. Требования к надёжности

Время непрерывной работы без сбоев – не менее 14 дней.

Время восстановления после сбоя – не более 1 часа.

При отказе одного канала уведомлений (например, FCM) Подсистема должна автоматически использовать альтернативный канал (email), если он доступен для данного студента.

Сохранность данных при авариях обеспечивается штатными механизмами резервного копирования БД ИС УДС.

В журнале событий должны фиксироваться все сбои и их причины.

4.4.4. Требования по безопасности

Доступ к функциям управления подписками осуществляется только после аутентификации в ИС УДС.

Персональные данные студентов обрабатываются в соответствии с Федеральным законом № 152-ФЗ «О персональных данных».

Передача данных по внешним каналам (FCM, SMTP, Telegram API) осуществляется по защищённым протоколам (HTTPS, TLS).

Доступ к логам и настройкам Подсистемы должен быть ограничен администратором (используется существующая система авторизации ИС УДС или отдельные учётные записи).

4.4.5. Требования к эргономике и технической эстетике

Интерфейс управления подписками должен быть интуитивно понятен, соответствовать общему стилю ИС УДС.

Уведомления должны быть краткими, информативными, не содержать избыточных данных. Рекомендуемый объём: не более 200 символов для push-уведомлений.

В тексте уведомлений не допускается использование нецензурной лексики, оскорбительных выражений.

4.4.6. Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов Подсистемы

Подсистема не требует специального обслуживания, кроме периодического контроля логов и своевременного обновления программного обеспечения (библиотек, сертификатов).

При обнаружении ошибок администратор должен зафиксировать их и, при необходимости, обратиться к Разработчику для исправления.

Резервное копирование данных (логов, настроек) выполняется в рамках общего резервирования сервера.

4.4.7. Требования к защите информации от несанкционированного доступа

Доступ к таблице `notification_subscriptions` и другим данным Подсистемы должен быть ограничен на уровне СУБД: только приложение Подсистемы имеет права на чтение/запись, администратор – через привилегированный доступ.

Пароли и ключи доступа к внешним сервисам (FCM, SMTP, Telegram) должны храниться в защищённом хранилище (например, переменные окружения, зашифрованный конфигурационный файл) и не передаваться в открытом виде.

Логи не должны содержать персональные данные в открытом виде (идентификаторы студентов допустимы, но без ФИО и контактной информации).

4.4.8. Требования по сохранности информации при авариях

В случае аварийного завершения работы Подсистемы неотправленные уведомления должны быть сохранены и обработаны после перезапуска.

Для этого используется механизм очередей (например, в базе данных или в Celery), который гарантирует, что каждое уведомление будет отправлено хотя бы один раз.

При потере соединения с внешними сервисами уведомления помещаются в очередь и повторяются с увеличивающимся интервалом.

4.4.9. Требования к защите от влияния внешних воздействий

Программное обеспечение Подсистемы должно быть устойчиво к скачкам напряжения, кратковременным отключениям питания (при использовании источников бесперебойного питания на сервере).

Сетевое оборудование должно обеспечивать защиту от DDoS-атак на уровне провайдера или организации.

Рекомендуется использование файервола для ограничения доступа к серверу только необходимыми портами.

4.4.10. Требования к патентной чистоте и патентоспособности

Разработка должна выполняться с использованием свободно распространяемого и лицензионно чистого программного обеспечения (Open Source или с соответствующими лицензиями).

При использовании сторонних библиотек необходимо соблюдать условия их лицензий.

4.4.11. Требования по стандартизации и унификации

При разработке следует применять стандарты оформления кода (PEP 8 для Python, Java Code Conventions).

Использовать унифицированные форматы данных (JSON для API).

Документацию оформлять в соответствии с ЕСПД (ГОСТ 19.XXX) и ГОСТ 7.32-2017.

Именованье переменных, функций, классов должно быть осмысленным и соответствовать принятым в проекте соглашениям.

4.4.12. Дополнительные требования

Подсистема должна поддерживать локализацию (русский язык) с возможностью добавления других языков в будущем.

Предусмотреть возможность отключения напоминаний в праздничные и нерабочие дни (настраивается администратором через файл конфигурации или веб-интерфейс).

В уведомлениях рекомендуется указывать ссылку на страницу с подробным расписанием (при наличии в ИС УДС).

При отправке уведомлений через Telegram рекомендуется использовать кнопки для быстрого перехода к расписанию (inline-клавиатура) – опционально.

5. Состав и содержание работ по созданию Подсистемы

Разработка Подсистемы автоматического напоминания о занятиях осуществляется поэтапно в соответствии с календарным планом, приведённым в таблице 1.

Таблица 1 – Календарный план работ по созданию Подсистемы

Этап	Содержание работ	Сроки выполнения	Ожидаемый результат
1. Формирование требований и разработка ТЗ	1.1. Анализ предметной области, изучение существующей ИС «Умный дневник студента». 1.2. Сбор и уточнение функциональных и нефункциональных требований. 1.3. Разработка и согласование технического задания (настоящий документ).	01.03.2026 – 10.03.2026	Утверждённое техническое задание.
2. Проектирование Подсистемы	2.1. Разработка архитектуры Подсистемы, определение состава модулей. 2.2. Проектирование структуры базы данных (таблица подписок). 2.3. Разработка схемы взаимодействия модулей и внешних сервисов (API, FCM, SMTP, Telegram). 2.4. Создание макетов интерфейсов (страницы управления подписками в личном кабинете).	11.03.2026 – 20.03.2026	Архитектурная спецификация, схема БД, макеты интерфейсов.
3. Разработка программного обеспечения	3.1. Реализация модуля сбора и анализа расписания. 3.2. Реализация модуля управления подписками. 3.3. Реализация модуля отправки уведомлений (push, email, Telegram). 3.4. Реализация модуля логирования и отчётности. 3.5. Интеграция модулей с ИС УДС (API, БД).	21.03.2026 – 10.04.2026	Работоспособные модули Подсистемы, исходный код.
4. Тестирование и отладка	4.1. Модульное тестирование каждого компонента. 4.2. Интеграционное тестирование взаимодействия модулей и внешних сервисов. 4.3. Функциональное тестирование согласно требованиям раздела 4. 4.4. Нагрузочное тестирование (имитация отправки большого количества уведомлений). 4.5. Исправление выявленных ошибок.	11.04.2026 – 20.04.2026	Протокол тестирования, исправленная версия ПО.

Продолжение таблицы 1

Этап	Содержание работ	Сроки выполнения	Ожидаемый результат
5. Подготовка документации	5.1. Разработка руководства администратора по настройке и эксплуатации. 5.2. Разработка инструкции для студентов по управлению подписками. 5.3. Оформление отчётной документации (пояснительная записка, приложения).	21.04.2026 – 25.04.2026	Комплект документации (электронный и бумажный экземпляры).
6. Опытная эксплуатация и сдача работы	6.1. Развёртывание Подсистемы на тестовом сервере. 6.2. Проведение опытной эксплуатации с участием ограниченного числа пользователей (например, учебной группы). 6.3. Сбор замечаний и их устранение. 6.4. Демонстрация Подсистемы Заказчику (преподавателю), подписание акта приёмки.	26.04.2026 – 30.04.2026	Акт приёмки выполненных работ, отчёт об опытной эксплуатации.

6. Порядок разработки Подсистемы

6.1. Порядок организации разработки Подсистемы

Разработка Подсистемы осуществляется в соответствии с календарным планом, приведённым в разделе 5 настоящего ТЗ. Организация работ включает:

- взаимодействие Разработчика с Заказчиком на всех этапах создания Подсистемы;
- регулярное представление промежуточных результатов Заказчику для согласования;
- ведение версионности разрабатываемого программного обеспечения с использованием системы контроля версий Git;
- документирование всех этапов разработки.

6.2. Перечень документов и исходных данных при разработке Подсистемы

Для разработки Подсистемы используются следующие документы и исходные данные:

- техническое задание на создание Подсистемы (настоящий документ);
- документация на существующую информационную систему «Умный дневник студента» (описание архитектуры, базы данных, API);
- стандарты и нормативные документы, указанные в разделе 1.8;
- данные о расписании учебных занятий, предоставляемые Заказчиком (в электронном виде);
- информация о группах, студентах и преподавателях из базы данных ИС УДС (доступ через API).

6.3. Перечень документов, предъявляемых по окончании соответствующих этапов работ

Таблица 2 – Перечень документов

Этап	Предъявляемые документы
1. Формирование требований и разработка ТЗ	- Утверждённое техническое задание.
2. Проектирование Подсистемы	- Архитектурная спецификация; - Схема базы данных; - Макеты интерфейсов (электронный вид).
3. Разработка ПО	- Исходный код модулей; - Исполняемые файлы (при необходимости).
4. Тестирование и отладка	- Протокол тестирования; - Исправленная версия ПО.
5. Подготовка документации	- Комплект эксплуатационной документации (руководство администратора, инструкция пользователя).
6. Опытная эксплуатация и сдача работы	- Отчёт об опытной эксплуатации; - Акт приёмки выполненных работ.

6.4. Порядок проведения экспертизы технической документации

Экспертиза технической документации проводится Заказчиком на этапе приёмки результатов каждого этапа. Замечания и предложения оформляются письменно или фиксируются в протоколе согласования. Разработчик обязан устранить замечания в согласованные сроки (не более 3 рабочих дней) и представить доработанные материалы.

6.5. Перечень макетов, порядок их разработки, изготовления, испытаний, необходимость разработки на них документации, программы и методик испытаний

Разработка макетов на этапе проектирования не предусмотрена, так как создаётся программный продукт. Допускается создание прототипов интерфейса для согласования с Заказчиком, но они не являются обязательными.

6.6. Порядок разработки, согласования и утверждения плана совместных работ по разработке Подсистемы

План совместных работ (календарный план) определён в разделе 5 настоящего ТЗ. Корректировка плана возможна по согласованию сторон при возникновении объективных причин. Изменения оформляются дополнительным соглашением или протоколом.

6.7. Порядок разработки, согласования и утверждения программы работ по стандартизации

Программа работ по стандартизации не разрабатывается, так как Подсистема создаётся в рамках учебного процесса и не подлежит обязательной сертификации. При разработке используются действующие стандарты (ГОСТ, ЕСПД), указанные в разделе 1.8.

6.8. Требования к гарантийным обязательствам Разработчика

В рамках учебного проекта гарантийные обязательства не предусмотрены. Однако Разработчик обязуется устранять ошибки, выявленные в процессе опытной эксплуатации, в течение 5 рабочих дней после их обнаружения (при наличии технической возможности).

6.9. Порядок проведения технико-экономической оценки разработки Подсистемы

Технико-экономическая оценка не проводится, так как работа выполняется в учебных целях без финансирования. Допускается приблизительная оценка трудозатрат (в часах) для отчёта.

6.10. Порядок разработки, согласования и утверждения программы метрологического обеспечения, программы обеспечения надежности, программы эргономического обеспечения

Указанные программы не разрабатываются ввиду отсутствия требований к метрологическому обеспечению (п. 4.3.8) и достаточности общих требований к надёжности и эргономике (п. 4.4.3, 4.4.5).

7. Порядок контроля и приемки Подсистемы

7.1. Виды, состав и методы испытаний Подсистемы и ее составных частей

Испытания Подсистемы проводятся с целью проверки соответствия требованиям настоящего ТЗ. Предусматриваются следующие виды испытаний:

7.1.1. Модульное тестирование

Проводится Разработчиком в процессе создания каждого модуля.

Цель: проверка работоспособности отдельных функций.

Метод: запуск модуля с тестовыми данными, сравнение фактических результатов с ожидаемыми.

7.1.2. Итерационное тестирование

Проводится после объединения всех модулей в единую Подсистему.

Цель: проверка корректности взаимодействия модулей между собой и с внешними сервисами (API ИС УДС, FCM, SMTP, Telegram).

Метод: выполнение сквозных сценариев (например, сбор расписания → формирование списка получателей → отправка уведомлений → запись в лог).

7.1.3. Функциональное тестирование

Проводится на завершающем этапе разработки.

Цель: проверка реализации всех функций, указанных в разделе 4.2.

Метод: выполнение тестовых сценариев, охватывающих все функции, с фиксацией результатов в протоколе.

7.1.4. Нагрузочное тестирование

Проводится для оценки производительности Подсистемы.

Цель: проверка способности обрабатывать ожидаемое количество уведомлений (до 100 в секунду) без существенных задержек.

Метод: имитация отправки большого числа уведомлений с помощью специальных скриптов, измерение времени отклика и количества ошибок.

7.1.5. Опытная эксплуатация

Проводится после успешного завершения функционального и нагрузочного тестирования.

Цель: проверка работы Подсистемы в реальных условиях с участием ограниченного числа пользователей (например, одной учебной группы).

Длительность: не менее 5 рабочих дней.

В процессе опытной эксплуатации собираются замечания пользователей, анализируются логи, при необходимости вносятся корректировки.

7.2. Общие требования к приемке работ, порядок согласования и утверждения приемочной документации

Приёмка работ осуществляется поэтапно в соответствии с календарным планом (раздел 5).

По окончании каждого этапа Разработчик представляет Заказчику результаты, указанные в п. 6.3.

Заказчик в течение 3 рабочих дней проверяет представленные материалы и при отсутствии замечаний подписывает акт сдачи-приёмки этапа (может быть оформлен как отдельный документ или отметка в ведомости).

При наличии замечаний они фиксируются в протоколе, и Разработчик устраняет их в согласованные сроки.

Окончательная приёмка всей Подсистемы осуществляется по завершении опытной эксплуатации. Разработчик предоставляет:

- отчёт об опытной эксплуатации (с указанием выявленных недостатков и их устранения);
- полный комплект документации (п. 6.3);
- акт приёмки выполненных работ (подписывается Заказчиком и Разработчиком).

7.3. Статус приемочной комиссии

Для приёмки результатов учебного проекта создаётся внутренняя приёмочная комиссия в составе:

- председатель комиссии;
- Разработчик;
- при необходимости – наблюдатели.

Статус комиссии – внутренняя (ведомственная). Протокол приёмки утверждается заведующим кафедрой (или преподавателем) и является основанием для оценки работы.

8. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу Подсистемы в действие

Для обеспечения готовности объекта автоматизации (процесса информирования студентов о расписании) к вводу Подсистемы в действие необходимо выполнить следующий комплекс мероприятий:

8.1. Организационные мероприятия

Назначить администратора Подсистемы из числа сотрудников кафедры или ИТ-отдела, ответственного за первоначальную настройку, мониторинг и эксплуатацию.

Издать приказ (распоряжение) о вводе Подсистемы в опытную эксплуатацию с указанием сроков и ответственных лиц.

Обеспечить доступ Разработчика к серверу ИС УДС и внешним сервисам (FCM, SMTP-сервер, Telegram Bot API) на время опытной эксплуатации (при необходимости).

8.2. Технические мероприятия

Проверить готовность серверного оборудования и программного обеспечения ИС УДС к установке модулей Подсистемы (наличие необходимых библиотек, свободного дискового пространства, прав доступа).

Обеспечить наличие и корректность учётных записей для внешних сервисов (ключи FCM, параметры SMTP, токен Telegram-бота).

Выполнить резервное копирование базы данных ИС УДС перед началом опытной эксплуатации.

8.3. Информационные мероприятия

Провести анализ полноты и качества данных в таблицах ИС УДС, необходимых для работы Подсистемы (students, groups, schedule). При необходимости выполнить их очистку или дополнение.

Подготовить тестовые наборы данных для проверки функциональности Подсистемы (например, расписание на несколько дней для выбранной группы).

8.4. Кадровые мероприятия и обучение персонала

Провести инструктаж администратора Подсистемы по настройке и эксплуатации (1 час). Инструктаж проводит Разработчик на основе разработанного руководства администратора.

Подготовить и опубликовать инструкцию для студентов по управлению подписками (в электронном виде, например, в разделе «Помощь» ИС УДС).

Оповестить студентов (через старост или объявление на портале) о возможности использования функции напоминаний и порядке её настройки.

8.5. Создание условий функционирования

Убедиться, что сервер ИС УДС соответствует минимальным техническим требованиям (п. 4.3.3).

Обеспечить стабильное интернет-соединение для доступа к внешним сервисам уведомлений.

Проверить работу системы резервного копирования для сохранности логов и данных подписок.

9. Требования к документированию

9.1.1. Перечень подлежащих разработке документов

В процессе создания Подсистемы должна быть разработана следующая документация:

- Техническое задание на создание Подсистемы (настоящий документ).
- Пояснительная записка (эскизный проект) – включает архитектуру, схемы модулей, описание базы данных.
- Руководство администратора – содержит инструкции по установке, настройке, мониторингу и устранению неполадок Подсистемы.
- Руководство пользователя (инструкция для студентов) – описывает порядок управления подписками на уведомления (включение/отключение, выбор каналов и времени упреждения).
- Программа и методика испытаний – содержит описание тестовых сценариев, методики проведения функционального и нагрузочного тестирования.
- Протокол испытаний – оформляется по результатам тестирования (этап 4).
- Отчёт об опытной эксплуатации – включает анализ работы Подсистемы, выявленные замечания и их устранение.
- Акт приёмки выполненных работ – подписывается Заказчиком и Разработчиком по окончании работ.

9.2. Вид представления и количество документов

Документация предоставляется в электронном виде (форматы: PDF, DOCX, ODT) на электронном носителе (CD/DVD или флеш-накопитель) – 1 экземпляр.

На бумажном носителе (при необходимости) – 1 экземпляр (сброшюрованный).

Исходный код Подсистемы предоставляется в электронном виде (архив с файлами) с комментариями.

9.3. Требования по использованию стандартов при разработке документов

Документация должна оформляться в соответствии с требованиями:

- ГОСТ 7.32–2017 «Отчёт о научно-исследовательской работе. Структура и правила оформления»;
- ГОСТ 19.106–78 «Единая система программной документации. Требования к программным документам, выполненным печатным способом»;
- ГОСТ 2.105–95 «Единая система конструкторской документации. Общие требования к текстовым документам».

Схемы алгоритмов и диаграммы выполняются в соответствии с ГОСТ 19.701–90 (ИСО 5807–85) «Единая система программной документации. Схемы алгоритмов, программ, данных и систем».

Библиографические ссылки и список использованных источников оформляются по ГОСТ Р 7.0.100–2018 «Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Библиографическое описание».

10. Источники разработки

При разработке Подсистемы использованы следующие документы и информационные материалы:

1. ГОСТ 34.602–2020 «Информационные технологии. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы». – М.: Российский институт стандартизации, 2021.
2. ГОСТ 7.32–2017 «Отчёт о научно-исследовательской работе. Структура и правила оформления». – М.: Стандартинформ, 2018.
3. ГОСТ 19.701–90 (ИСО 5807–85) «Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения». – М.: Стандартинформ, 2010.
4. ГОСТ Р 7.0.100–2018 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления». – М.: Стандартинформ, 2018.
5. Документация на существующую информационную систему «Умный дневник студента» (описание базы данных, API, архитектуры), предоставленная Заказчиком.
6. Материалы по интеграции с Firebase Cloud Messaging: документация Firebase, руководства по использованию Firebase Admin SDK. – URL: <https://firebase.google.com/docs/cloud-messaging> (дата обращения: 25.02.2026).
7. Материалы по отправке электронной почты через SMTP: документация по протоколу SMTP, библиотеки для Python (smtplib). – URL: <https://docs.python.org/3/library/smtplib.html> (дата обращения: 25.02.2026).
8. Документация Telegram Bot API: описание методов, примеры использования. – URL: <https://core.telegram.org/bots/api> (дата обращения: 25.02.2026).