

**хекслет
колледж**

«Колледж Хекслет» ЖШС
Мекен жайы: Қазақстан
Республикасы, Алматы қ.,
Назарбаев даңғылы, 103
e-mail: itcollege@hexly.kz



**хекслет
колледж**

ТОО «Колледж Хекслет»
Адрес: Республика
Казахстан, г. Алматы,
проспект Назарбаева, 103
e-mail: itcollege@hexly.kz

«УТВЕРЖДЕНЫ»

**Решением УМС
Председатель**

_____ Д.Калиаскаров
«___» _____, протокол _____

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОФОРМЛЕНИЮ
ОТЧЕТА ПО УЧЕБНОЙ ПРАКТИКЕ
ПО ПМЗ «РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ»
СПЕЦИАЛЬНОСТИ:
06130100 – ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ (ПО ВИДАМ)**

Рассмотрены
на заседании ПЦК «» Протокол № «___» от «___» ____ 20____ г.

Алматы

1. Общие положения

1.1. Отчёт по учебной практике является самостоятельной работой студента, выполняемой в процессе прохождения практики по профессиональному модулю ПМ3 «Разработка модулей программного обеспечения для компьютерных систем» под руководством преподавателя.

1.2. Цель подготовки отчёта по практике:

- систематизация, закрепление и расширение теоретических знаний и практических навыков по модулю ПМ3;
- применение полученных знаний при решении конкретных учебно-производственных задач (разработка веб-приложения с использованием облачной баз данных Firebase);
- развитие навыков самостоятельной работы с облачными технологиями, документацией, программным обеспечением и техническими заданиями;
- формирование умений проектирования структуры базы данных, разработки архитектуры программного продукта, описания моделей и алгоритмов;
- проверка готовности студента к выполнению профессиональных задач в условиях, приближённых к деятельности fullstack-разработчика.

1.3. Отчёт по учебной практике представляет собой обобщение результатов самостоятельной работы студента при выполнении практического проекта, включающего проектирование, разработку и описание веб-приложения на базе Firebase (Firestore и Authentication).

1.4. Отчёт по практике должен отвечать следующим требованиям:

- содержать анализ и описание архитектуры разрабатываемого веб-приложения;
- отражать этапы реализации проекта в соответствии с техническим заданием и выбранной темой;
- содержать профессионально ориентированные элементы программной документации (схему БД, описание классов, структуру API, правила безопасности);
- включать обоснованные выводы о выполненной работе и приобретённых профессиональных компетенциях;
- прилагать материалы, подтверждающие выполнение практики: схемы базы данных, таблицы, скриншоты интерфейсов, а также программный код в приложении.

2. Содержание и структура отчета

2.1. Отчёт по учебной практике должен отвечать установленным требованиям не только по содержанию, но и по логике изложения материала.

2.2. Структурными элементами отчёта являются:

- титульный лист (1 стр.);
- содержание (1 стр.);
- введение (1–2 стр.);
- характеристика базы практики (1–2 стр.);
- постановка задачи (техническое задание) (2–3 стр.);
- проектирование базы данных (схема Firestore) (2–3 стр.);
- описание внешней структуры сайта (1–2 стр.);
- описание внутренней структуры сайта (2–3 стр.);
- практическая часть (основные этапы реализации проекта: 8–12 стр.);
- заключение (1–2 стр.);
- список использованных источников (не менее 10);
- приложения (листинг программы, скриншоты интерфейсов, схемы).

2.3. Титульный лист является первой страницей отчета и служит источником информации, необходимой для обработки и поиска документа.

2.4. На титульном листе приводятся следующие сведения:

- наименование образовательной организации;
- вид работы: отчет по учебной практике;
- тема отчёта: «Разработка веб-приложения "[Название проекта]" с использованием Firebase»;
- профессиональный модуль: ПМ3 «Разработка модулей программного обеспечения для компьютерных систем»;
- шифр и наименование специальности, квалификации;
- справа слово «выполнил(а)» указывается фамилия и инициалы обучающегося, курс и группа;
- двумя строками ниже пишется «Допущен(-а) к защите» с указанием места для даты, а также «Преподаватель», его фамилия и инициалы, место для подписи. Правее пишется строка «Заштил(-а) с оценкой:» с указанием места для буквы, цифрового эквивалента, а также балла.
- город, год.

2.5. Содержание отчёта включает все разделы, подразделы, заключение, список источников и приложения.

2.6. Введение (начинается на третьей странице отчета), содержит обоснование актуальности выбранной темы проекта, цель и задачи практики, описание ожидаемых результатов, а также методы работы. Объем введения составляет 1-2 страницы.

2.7. Проектирование базы данных (Firestore Schema)

Глава "Проектирование базы данных" содержит:

1. Выбор темы проекта (из предложенного списка) и краткое описание предметной области.
2. Визуальная схема структуры Firestore (обязательный элемент):

3. Все коллекции с названиями
4. Структура документов (основные поля с типами данных)
5. Связи между коллекциями
6. Составные ID (если используются)
7. Примеры заполнения документов

2.8. Проектирование сайта. Глава содержит два пункта:

1. Внешняя структура сайта

Отражает логику навигации между страницами приложения. Может быть представлена в виде схемы или таблицы.

Основные страницы сайта (общие для всех тем):

1. Главная страница (index.html) – каталог/лента основных сущностей
2. Страница детального просмотра (detail.html?id=ID) – полная информация об элементе
3. Личный кабинет (profile.html) – профиль пользователя, история действий
4. Функциональная страница (cart.html / bookings.html / my.html) – активные действия
5. Авторизация/Регистрация (auth.html) – формы входа и регистрации
6. Админ-панель (admin.html) – управление контентом (для пользователей с ролью admin)

Таблица 2 – Внешняя структура сайта

Страница	Основное назначение	Доступные переходы
Главная	Каталог/лента	На детальную страницу, в личный кабинет, в авторизацию
Детальная	Полная информация	В функциональную страницу (корзина/бронирование), в каталог
Личный кабинет	Профиль и история	Просмотр деталей, редактирование профиля
Авторизация	Вход/Регистрация	В каталог, восстановление пароля

2. Внутренняя структура сайта.

Перечень файлов и папок проекта: HTML-страницы, JS-модули, CSS, изображения. Даётся краткое описание функций каждого модуля и их взаимодействия.

Основные директории и файлы проекта:

1. index.html, detail.html, profile.html, cart.html, auth.html, admin.html – HTML-шаблоны страниц
2. /css/styles.css – стили оформления
3. /js/ – JavaScript модули (Firebase-интеграция)
4. /images/ – изображения

Назначение JavaScript-модулей:

1. firebase-config.js – инициализация Firebase, конфигурация проекта
2. auth-service.js – работа с Firebase Authentication (регистрация, вход, выход)
3. db-service.js – работа с Firestore (CRUD операции, запросы)

4. [тема]-service.js – сервис для работы с основными сущностями (product-service.js, recipe-service.js и т.д.)
5. ui-service.js – отрисовка интерфейса, обработка событий
6. admin-service.js – функции админ-панели
7. utils.js – общие утилиты, валидация, форматирование

Таблица 3 – Внутренняя структура сайта

Файл	Назначение	Взаимодействие
firebase-config.js	Инициализация Firebase	Все модули
auth-service.js	Аутентификация пользователей	db-service.js, ui-service.js
db-service.js	Работа с Firestore	Все сервисы
[тема]-service.js	Логика предметной области	db-service.js, ui-service.js

2.9. Практическая часть

Глава отражает последовательность выполнения проекта и показывает, как студент применил знания и навыки в ходе практики. Здесь необходимо описывать действия, результаты и выводы, избегая избыточного кода в основном тексте (код выносится в приложения).

1. Настройка Firebase проекта

- создание проекта в Firebase Console;
- настройка Authentication (метод входа: email/password);
- настройка Firestore Database в тестовом режиме;
- получение конфигурации Firebase для веб-приложения;
- создание файла firebase-config.js с инициализацией.

2. Создание схемы базы данных и тестовых данных

- проектирование структуры коллекций согласно выбранной теме;
- создание тестовых данных через Firebase Console или скрипт инициализации;
- настройка индексов для сложных запросов;
- создание правил безопасности (Firestore Rules).

3. Реализация системы аутентификации

- создание страницы auth.html с формами регистрации и входа;
- реализация регистрации пользователя (createUserWithEmailAndPassword);
- реализация входа в систему (signInWithEmailAndPassword);
- обработка восстановления пароля;
- сохранение состояния аутентификации (onAuthStateChanged);
- создание документа пользователя в Firestore при регистрации.

4. Разработка главной страницы с пагинацией

- создание функции загрузки данных из Firestore с использованием .limit();
- реализация пагинации через .startAfter();
- отображение карточек элементов с основными данными;

- реализация кнопки "Показать еще";
- добавление индикатора загрузки.

5. Реализация поиска и фильтрации

- создание поля поиска на главной странице;
- реализация поиска по Firestore (использование составных индексов);
- добавление фильтров по категориям/параметрам;
- реализация сортировки по различным критериям.

6. Страница детального просмотра

- получение ID элемента из URL (URLSearchParams);
- загрузка полной информации о элементе из Firestore;
- отображение всех связанных данных;
- реализация основного действия (добавить в корзину/забронировать/оценить);
- проверка прав доступа для действий.

7. Система взаимодействия (отзывы/оценки/бронирования)

- создание формы для добавления отзывов/оценок;
- проверка: можно ли оставить отзыв (только один, только после покупки и т.д.);
- реализация системы оценок с обновлением среднего рейтинга;
- отображение отзывов с пагинацией.

8. Функциональная страница (корзина/избранное/бронирования)

- создание страницы для управления активными действиями;
- реализация добавления/удаления/изменения количества;
- подсчет итоговой суммы/статистики;
- оформление основного действия (оформить заказ/подтвердить бронь);
- очистка после завершения действия.

9. Личный кабинет пользователя

- отображение истории действий пользователя;
- реализация детального просмотра каждого действия;
- форма редактирования профиля (имя, контакты, настройки);
- отображение личной статистики (опционально);
- режим реального времени для обновления статусов.

10. Реализация режима реального времени

- использование onSnapshot() для отслеживания изменений;
- реализация в одном или нескольких модулях (корзина, статусы заказов, новые элементы);
- обработка изменений и обновление интерфейса;
- отписка от слушателей при необходимости.

11. Админ-панель (для пользователей с ролью admin)

- проверка роли пользователя перед доступом;
- управление основными сущностями (CRUD операции);
- управление действиями пользователей (просмотр, изменение статусов);
- модерация контента (удаление некорректных отзывов);
- базовая статистика (опционально).

12. Оптимизация и правила безопасности

- использование `.select()` для выбора только необходимых полей;
- создание дублированных полей для оптимизации запросов;
- предварительное вычисление счетчиков (`reviewCount`, `averageRating`);
- настройка правил безопасности Firestore (разные права для `user/admin`);
- обработка ошибок и edge cases.

В тексте отчёта описывается последовательность действий, даются пояснения и вставляются скриншоты интерфейсов. Кодовые листинги (JavaScript, HTML, CSS) выносятся в приложения.

2.10. Заключение должно содержать оценку выполненной работы: что реализовано полностью, что требует доработки, какие навыки и компетенции студент приобрёл. Итоговые выводы о прохождении практики и достижении поставленных целей.

2.11. Список использованных источников содержит перечень разнообразных источников, расположенных в порядке появления их в тексте работы. Общее количество литературных источников – не менее 10.

Включаются:

1. Официальная документация Firebase
2. Учебные материалы по JavaScript и веб-разработке
3. Статьи и блоги по выбранной теме проекта
4. Примеры проектов на GitHub
5. Документация по используемым библиотекам

2.12. ПРИЛОЖЕНИЯ. В приложения выносятся:

1. Листинг программы (JS-модули, HTML-файлы, CSS)
2. Схема базы данных Firestore (полная версия)
3. Правила безопасности Firestore
4. Скриншоты всех реализованных страниц
5. Скриншоты Firebase Console (проект, данные, использование)
6. Примеры документов из Firestore

3. Порядок защиты отчета по учебной практике

3.1. В ходе защиты студент должен показать:

- понимание целей и задач практики по модулю ПМЗ;
- знание архитектуры своего проекта;
- умение объяснить структуру базы данных и выбор проектных решений;
- способность демонстрировать работу приложения;
- анализ достигнутых результатов.

3.2. Процедура защиты включает:

- демонстрацию работающего приложения (5-7 минут);
- презентацию схемы базы данных и архитектуры;
- ответы на вопросы преподавателя по реализации;
- показ Firebase Console с данными проекта.

3.3. Защита проводится в последний день практики. Преподаватель принимает отчёт, заслушивает выступление и выставляет итоговую оценку.

3.4. Общая схема доклада (7–10 минут):

- представление темы проекта и её актуальности (1 мин);
- демонстрация схемы базы данных (2 мин);
- показ основных функций приложения (3 мин);
- объяснение технических решений (оптимизация, безопасность) (2 мин);
- выводы и приобретённые навыки (1 мин).

4. Оценивание учебной практики

4.1. Общая система оценивания строится по 100-балльной шкале.

4.2. Критерии и распределение баллов:

1. Содержание и полнота отчёта – до 25 баллов:

- наличие всех обязательных разделов – 5 баллов;
- качество схемы базы данных (полнота, понятность, оптимизация) – 10 баллов;
- раскрытие внешней и внутренней структуры – 5 баллов;
- наличие иллюстраций и документации – 5 баллов.

2. Практическая реализация проекта – до 50 баллов:

- настройка Firebase и аутентификация – 10 баллов;
- работа с Firestore (CRUD, запросы) – 10 баллов;
- реализация основных функций по теме – 10 баллов;
- пагинация и поиск/фильтрация – 10 баллов;
- админ-панель и правила безопасности – 10 баллов.

3. Качество кода и архитектура – до 15 баллов:

- модульная структура, классы, сервисы – 5 баллов;
- обработка ошибок и edge cases – 5 баллов;
- оптимизация запросов (select, индексы, дублирование) – 5 баллов.

4. Защита отчёта – до 10 баллов:

- демонстрация работающего приложения – 5 баллов;
- ответы на вопросы, понимание архитектуры – 5 баллов.

4.3. Итоговая шкала оценки:

- 90–100 баллов (A, A-) – отчёт полный, проект реализован полностью с продвинутыми функциями, отличная архитектура, защита проведена уверенно.
- 75–89 баллов (B+, B, B-) – реализованы все основные функции, схема БД качественная, есть мелкие недочёты.
- 65–74 баллов (C+, C) – реализована большая часть функционала, схема БД присутствует, есть значительные недочёты.
- 50–64 баллов (C-, D+, D) – реализована только базовая функционал, схема БД схематичная, много недочётов.
- 0–49 баллов (F) – проект не соответствует требованиям, функционал не реализован.

4.4. Буквенная система оценки:

Оценка	Цифровой эквивалент	Процент содержания	Традиционная система
A	4	95-100	Отлично

A-	3,67	90-94	
B+	3,33	85-89	Хорошо
B	3	80-84	
B-	2,67	75-79	
C+	2,33	70-74	Удовлетворительно
C	2	65-69	
C-	1,67	60-64	
D+	1,33	55-59	
D	1	50-54	
F	0	0-49	Неудовлетворительно

4.5. При неудовлетворительной оценке студент имеет право повторно защищать работу после доработки в установленные сроки.

Приложение 1

Образец оформления титульного листа

ТОО «Колледж Хекслет»

ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ

**на тему: «Разработка прототипа интернет-магазина
_____” (учебный проект с использованием облачной
базы данных»**

по модулю: «Разработка модулей программного обеспечения для компьютерных систем»

Специальность: 06130100 – Программное обеспечение (по видам)

Квалификация: 4S06130105 – Техник информационных систем

Выполнил(-а): студент 3 курса
22 ТИС группы
Ким Артур Артурович

Допущен(-а) к защите «_____» 20_г.
Преподаватель: Бенли Р.А.

(подпись)

Зашитил(-а) с оценкой:

(буква) (циф.экв.) (балл)

Приложение 2
Образец содержания

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	2
1 ПОСТАНОВКА ЗАДАЧИ (ТЕХНИЧЕСКОЕ ЗАДАНИЕ)	3
1.1 Цель проекта	3
1.2 Функциональные требования	5
1.3 Нефункциональные требования	7
1.4 Критерии готовности	9
2 ПРОЕКТИРОВАНИЕ САЙТА	11
2.1 Проектирование базы данных	12
2.1 Внешняя структура сайта	13
2.2 Внутренняя структура сайта	15
3 ПРАКТИЧЕСКАЯ ЧАСТЬ РЕАЛИЗАЦИИ ПРОЕКТА	17
3.1 Настройка Firebase проекта	18
3.2 Создание схемы базы данных и тестовых данных	19
3.3 Реализация системы аутентификации	20
3.4 Разработка главной страницы с пагинацией	21
3.5 Реализация поиска и фильтрации	22
3.6 Страница детального просмотра	23
3.7 Система взаимодействия (отзывы/оценки/бронирования)	24
3.8 Функциональная страница (корзина/избранное/бронирования)	25
3.9 Личный кабинет пользователя	
3.10 Реализация режима реального времени	
3.11 Админ-панель	
3.12 Оптимизация и правила безопасности	
ЗАКЛЮЧЕНИЕ	26
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	27
ПРИЛОЖЕНИЯ	28