

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ОТЧЁТ
к лабораторной работе № 3

Студенты:

И.М. Косяк
В.Д. Богаченко
Д.А. Гринкевич

Проверил:

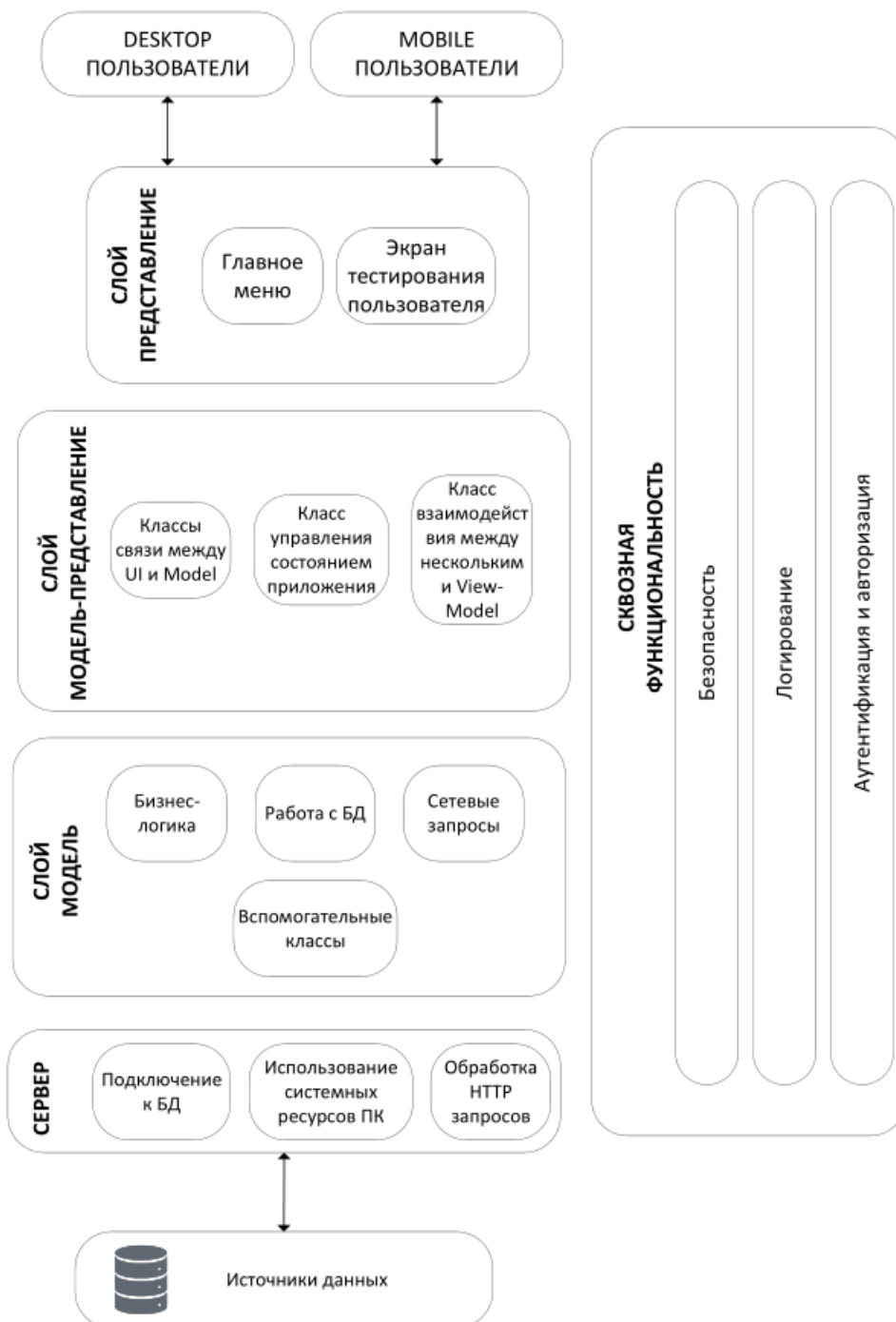
Е.В. Богдан

МИНСК 2024

1 АРХИТЕКТУРА TO BE

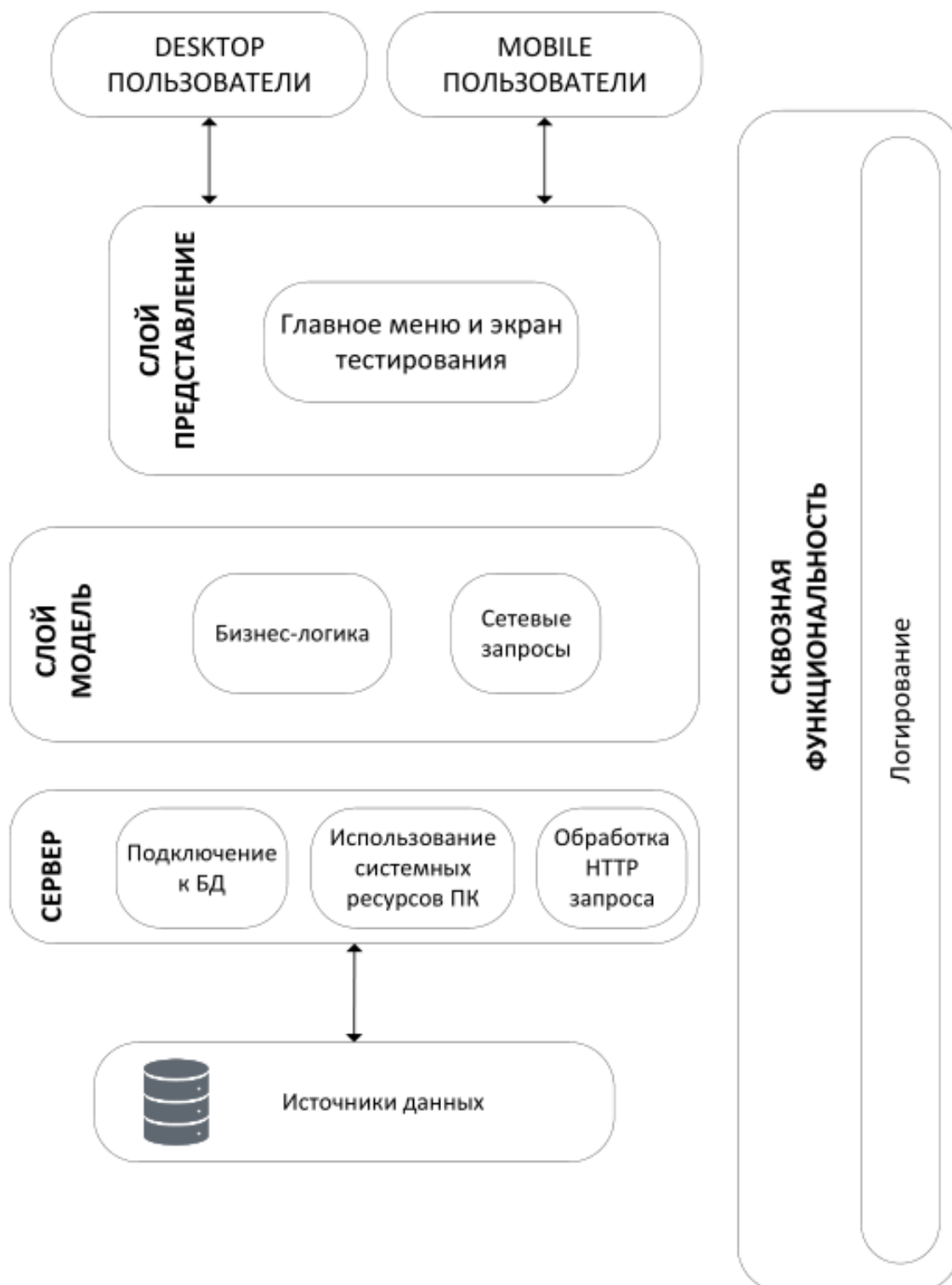
В рамках данной лабораторной работы мы провели сравнение двух архитектурных решений: "To Be" и "As Is". "To Be" представляет собой архитектуру нашего приложения, как мы видим ее в законченном варианте, а "As Is" - имеющуюся архитектуру на данный момент, полученную с помощью инструментов обратной инженерии нашего исходного кода.

Архитектура "To Be" представлена в документе "To be.pdf".



2 АРХИТЕКТУРА AS IS

Архитектура "As Is" представлена в документе "As is.pdf". Она включает следующие компоненты:



3 СРАВНЕНИЕ АРХИТЕКТУР

После проведения сравнения этих двух архитектурных решений и выявления отличий, мы пришли к следующим результатам:

1. Отличия между "To Be" и "As Is" архитектурами:

- В существующей архитектуре("As Is") отсутствует связующий слой ViewModel(модель-представление).

- В слое представления есть только один экран("As Is"), в "To Be" интерфейс разделен.

- В слое модели в архитектуре "To Be" есть работа с локальной БД в отличие от "As Is", в которой её вообще нет (только запросы на сервер с удаленной БД).

- На сервере в "to is" только 1 возможный GET запрос, когда в "To Be" их достаточно для реализации расширенного функционала.

- В слое модели в архитектуре "To Be" есть работа с локальной БД в отличие от "As Is", в которой её вообще нет (только запросы на сервер с удаленной БД).

- В сквозной функциональности в "As Is" не обеспечена безопасность передачи и хранения данных, а также код программы для мобильных устройств не обфусцирован.

- Архитектура "To Be" спроектирована с возможностью авторизации пользователей в отличие от "As Is" без регистрации.

2. Причины отличий:

- Невозможность полной реализации всех компонентов и функциональности за 1 раз, так как разработка идёт постепенно (реализованы главные экраны для тестирования функциональности и дальнейшего проектирования интерфейса).

- Изменение требований и приоритетов проекта, что привело к ограниченной реализации некоторых компонентов.

- Невозможно писать отдельную часть проекта(web, mobile, backend) без коммуникации и внесения изменений во все остальные части. (web и mobile создаются похоже, однако что там что там бывает невозможно сделать что-то, а также должны быть известны точки подключения к серверу). По этой же причине аутентификация пользователей перенесена на следующий спринт.

- Обеспечение безопасности данных находится не на первом месте, самое приоритетное это удобство взаимодействия пользователя

4 ВЫВОД

В реальных условиях бизнес-процессы могут отличаться от того, что описано в инструкциях или должностных инструкциях. Люди могут работать по-разному, и единой системы “AS IS” может не существовать. Поэтому описывать текущее состояние может быть сложно.

Индивидуальные подходы: Сотрудники могут выполнять одни и те же задачи по-разному.

Изменение требований и приоритетов: Проекты могут меняться, и приоритеты могут сдвигаться. Это может повлиять на выбор архитектуры. Гибкость и адаптивность к изменениям важны.