

**CENTRUL DE EXCELENȚĂ ÎN INFORMATICĂ ȘI
TEHNOLOGII INFORMAȚIONALE**

СПЕЦИАЛЬНОСТЬ «Programare și testarea produselor de program»

Examen

Номер билета - 25

Преподаватель: Gairunova Natalia

Учащийся: Vorobiov Serghei

Кишинев, 2025

1. Сценарий настольного приложения «Управление рестораном».

Создайте прототип настольного приложения в Figma, который позволит: добавлять и управлять меню ресторана; управлять ими; искать и фильтровать по названию ресторана, названию меню, заказам, блюдам, персоналу; аутентифицировать пользователей (главный администратор/менеджер ресторана); просматривать статистику.

Frame 1

Frame 2

Frame 3

Frame 4

Frame 5

Frame 7

Frame 8

№	Название ресторана	Адрес	Количество персонала	Активные меню
1	La Vista	ул. Штейнан ч. Маркса, 12	12	4
2	Vanya	ул. Ильинская, 45	0	3
3	Bella Italia	ул. Пушкина, 8	7	2
4	Food Point	ул. Бакунинская, 21	5	1
5	Bimbo	ул. Адмиралтейская, 33	6	0

Frame 6

Показатель	Значение
Всего заказов за день	24
Самое популярное блюдо	Пasta Карбонара
Самый активный ресторан	La Vista
Среднее время выполнения заказа	18 минут
Количество активных сотрудников	27
Количество активных столов	12

Frame 9

Добавьте заказ

Название
Value

Имя
Value

Save Cancel

Позиции: 0 из 0

1270 x 854

2. Функциональные требования к прототипу (25 puncte)

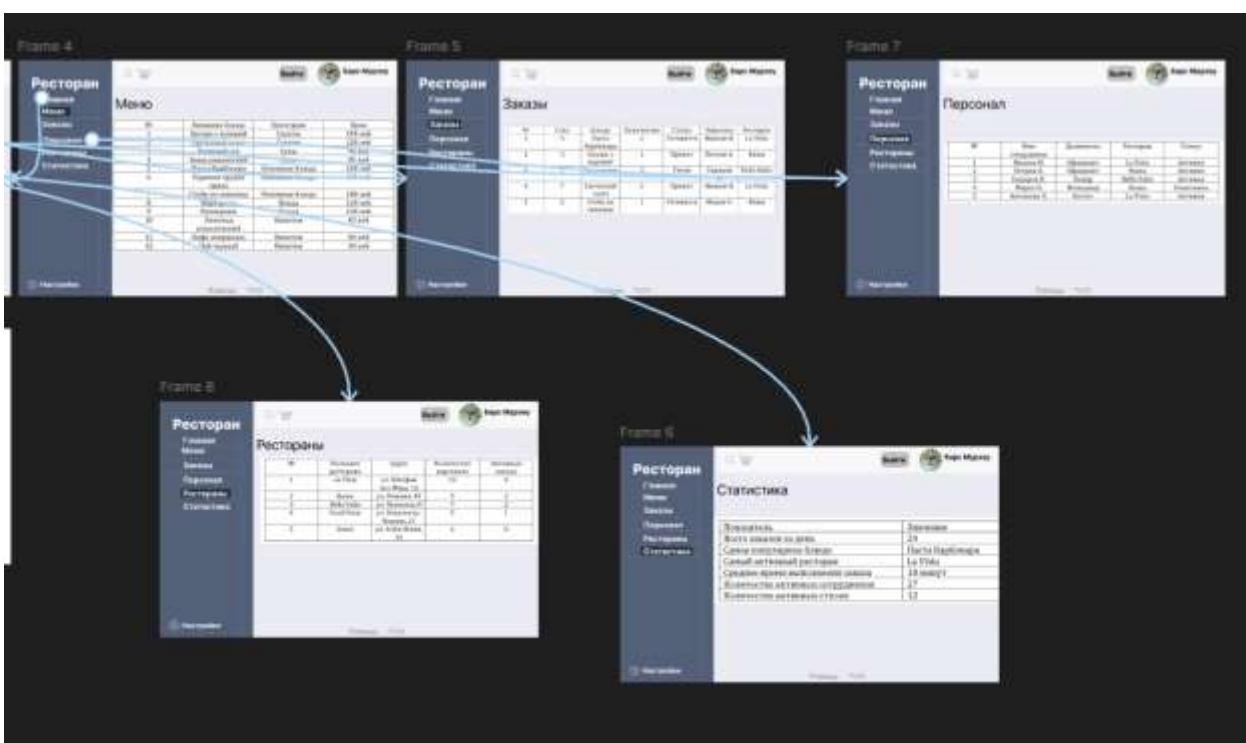
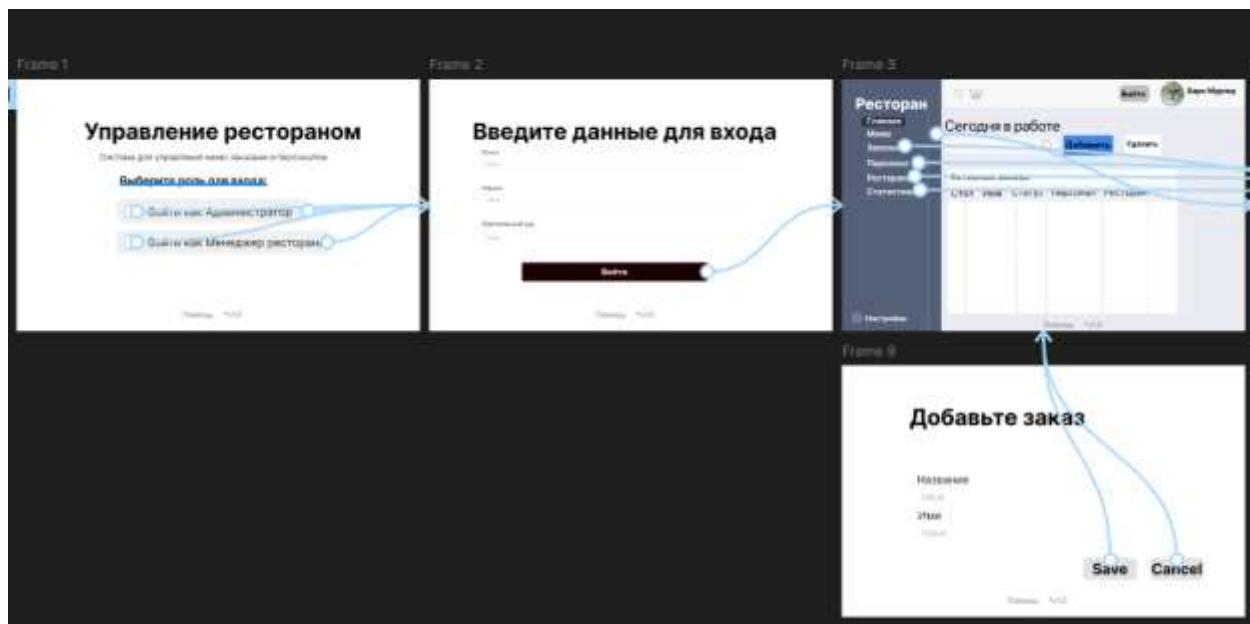
- а. Экран входа в систему с аутентификацией и проверкой.
- б. Главная панель управления:

 - Список активных заказов с указанием персонала и соответствующего ресторана, а также функции поиска по столу.
 - Кнопка для добавления/редактирования/удаления заказов.

- в. Страница добавления/изменения ресторана: форма с полями для ресторана, заказов и персонала.
- г. Страница статистики: ежедневные отчеты по меню и активным заказам.
- д. Навигация между экранами с помощью меню или кнопок..

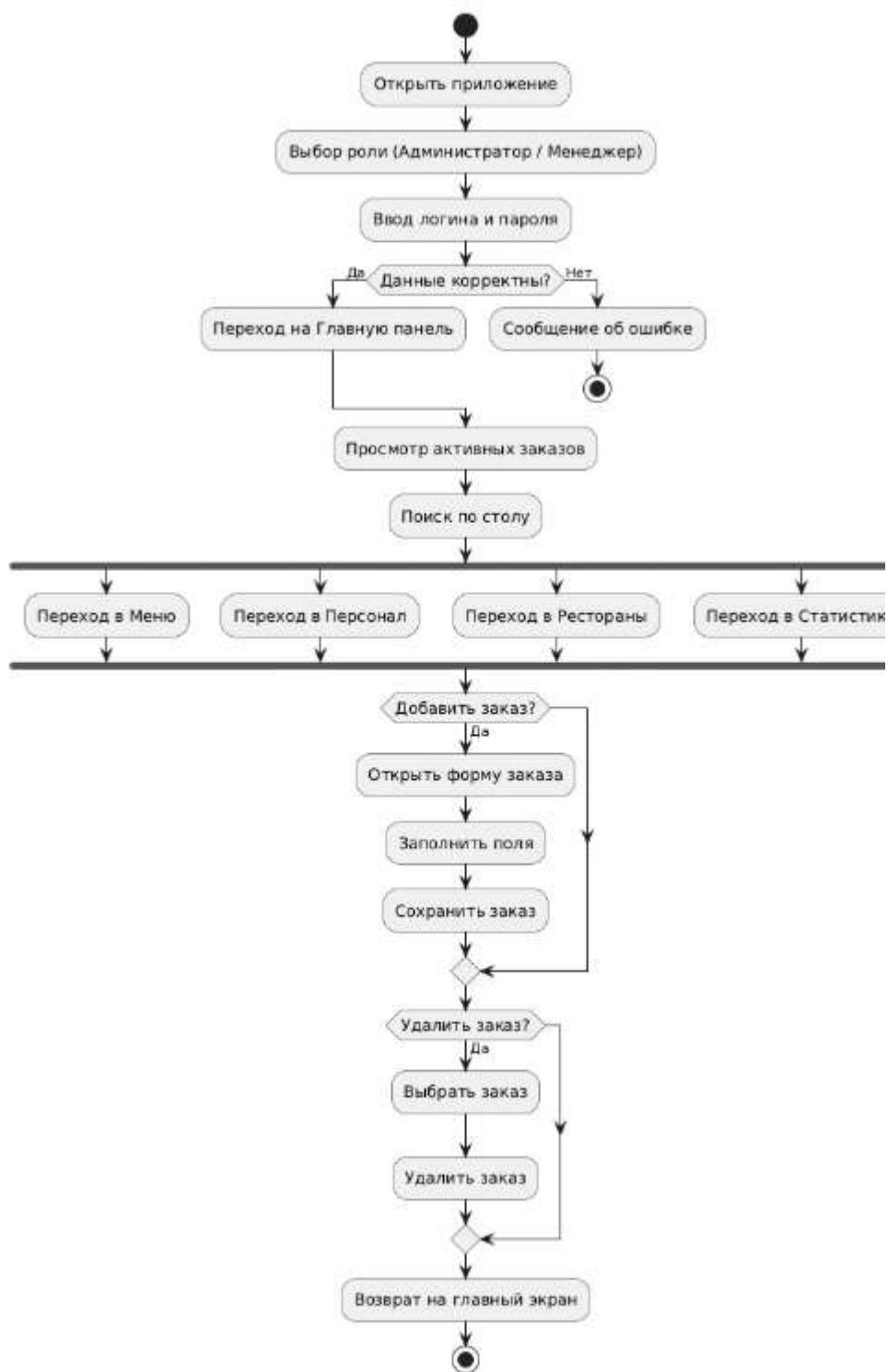
3. Нефункциональные требования (15 puncte)

- a)** Четкий и визуально последовательный интерфейс с использованием единой цветовой схемы и типографики.
- b)** Логичная и интуитивно понятная организация элементов пользовательского интерфейса (UI).
- c)** Простота отслеживания и навигации по потоку приложения.
- d)** Прототип должен быть полностью интерактивным в Figma (кликаемые экраны).



4. Создайте UML диаграммы: (15 пункте)

- a) Диаграмма вариантов использования (Use Case) для ролей администратора и менеджер.
- b) Диаграмма деятельности (Activity) для потока «Управление рестором».



```
1 @startuml
2 skinparam activityFontSize 13
3 skinparam activityArrowThickness 1
4
5 start
6
7 :Открыть приложение;
8 :Выбор роли (Администратор / Менеджер);
9
10 :Ввод логина и пароля;
11
12 if (данные корректны?) then (да)
13     :Переход на Главную панель;
14 else (нет)
15     :Сообщение об ошибке;
16     stop
17 endif
18
19 :Просмотр активных заказов;
20
21 :Поиск по столу;
22
23 fork
24     :Переход в Меню;
25 fork again
26     :Переход в Персонал;
27 fork again
28     :Переход в Рестораны;
29 fork again
30     :Переход в Статистику;
31 end fork
32
33 if (добавить заказ?) then (да)
34     :Открыть форму заказа;
35     :Заполнить поля;
36     :Сохранить заказ;
37 endif
38
39 if (Удалить заказ?) then (да)
40     :Выбрать заказ;
41     :Удалить заказ;
42 endif
43
44 :Возврат на главный экран;
45
46 stop
47
48 @enduml
49
```

```

@startuml
left to right direction
skinparam packageStyle rectangle
skinparam usecaseArrowThickness 1
skinparam usecaseFontSize 12
skinparam actorFontSize 12
skinparam usecaseBackgroundColor #EFEFEF

actor "Администратор" as Admin
actor "Менеджер" as Manager

rectangle "Система" {
    usecase "Вход в систему" as UC1
    usecase "Главная панель" as UC2
    usecase "Навигация между\постраницами" as UC3

    usecase "Просмотр заказов" as UC4
    usecase "Добавить заказ" as UC5
    usecase "Редактировать заказ" as UC6
    usecase "Удалить заказ" as UC7

    usecase "Просмотреть меню" as UC8
    usecase "Просмотреть персонал" as UC9
    usecase "Просмотреть рестораны" as UC10
    usecase "Просмотреть статистику" as UC11

    usecase "Добавить ресторан" as UC12 <<AdminOnly>>
    usecase "Изменить ресторан" as UC13 <<AdminOnly>>
    usecase "Удалить ресторан" as UC14 <<AdminOnly>>
}

" Общие функции
Admin --> UC1
Manager --> UC1

Admin --> UC2
Manager --> UC2

Admin --> UC3
Manager --> UC3

Admin --> UC4
Manager --> UC4

Admin --> UC5
Manager --> UC5

Admin --> UC6
Manager --> UC6

Admin --> UC7
Manager --> UC7

Admin --> UC8
Manager --> UC8

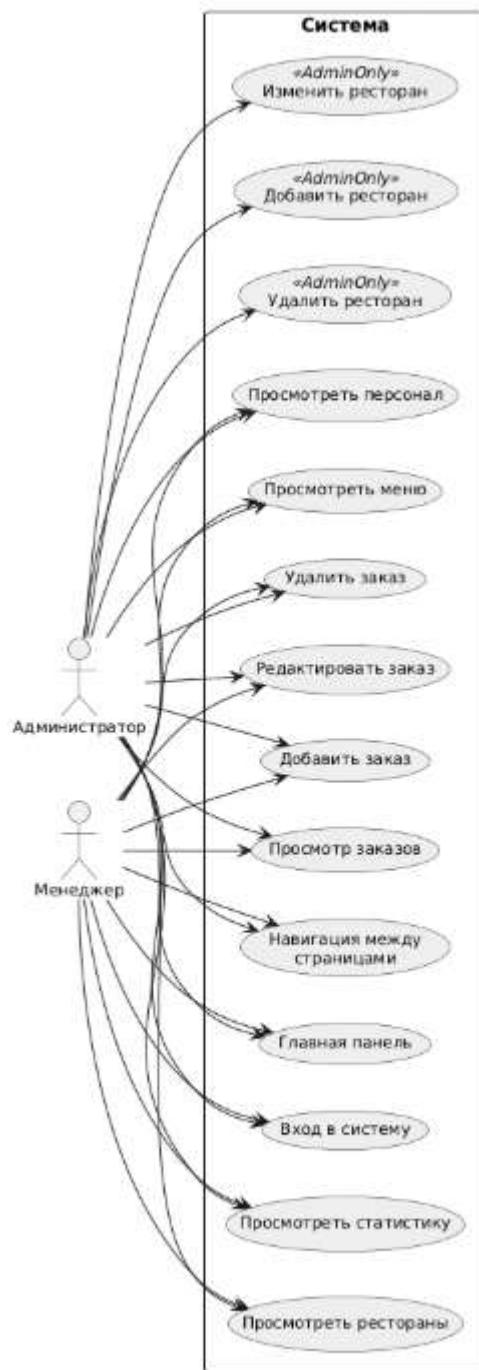
Admin --> UC9
Manager --> UC9

Admin --> UC10
Manager --> UC10

Admin --> UC11
Manager --> UC11

" Только администратор
Admin --> UC12
Admin --> UC13
Admin --> UC14

@enduml
[ ]
```



5. Технические требования и Git-поток (20 пункта)

1. Создание Git-репозитория с именем «Управление рестораном».
2. Сохранение всех экспортованных файлов Figma (PNG/SVG) и документации.
3. Четкие коммиты для каждого этапа: Login, Dashboard, Form, Statistics.
4. Публикация репозитория на GitHub.

<https://github.com/SergoMd/Upravlenie-restoranom>