



Проект **Foodie**

по дисциплине: Архитектура
цифрового предприятия

Команда: Воронина А., Гозун Е., Дьяконова М.,
Мишина А., Свищёва М.





План презентации

01

Описание предметной
области

02

Контекстная
диаграмма

03

Организационная
структура

04

Модель процессов
деятельности

05

Логическая модель
данных

06

Поддержка процессов
запросами

01 Описание предметной области



Тема проекта: Сервис по оптимизации пищевых отходов

Описание процесса:

Система для оптимизации пищевых отходов, которая интегрируется в рестораны.

Сервис Foodie работает с 16:00 до 24:00, предоставляя клиентам блюда из множества ресторанов разного уровня по сниженной цене.

Доставка блюд не осуществляется, клиенты забирают наборы самостоятельно.





01

Описание предметной области



● Масштаб проекта

Сервис сотрудничает с X числом ресторанов в г. Москва и обслуживает N уникальных пользователей приложения (сайта) Foodie.

● Границы проекта

Мобильное приложение (сайт) для клиентов с целью оформления заказов; база данных для администраторов ресторанов и сотрудников Foodie.



01

Описание предметной области



Финансовая модель

Выручка сервиса = 50% от разницы между ценой ресторана и ценой продажи.

Пример:

В бокс входит 2 блюда за 130 и 170 руб. (цена ресторана). Цена на бокс назначается менеджером Foodie и составляет 350 руб. (цена продажи).

$$350 - 130 - 170 = 50$$

Выручка сервиса = $50 * 0,5 = 25$ руб.



01

Описание предметной области



Временные ограничения

- Отправка заказа в ресторан осуществляется за 30 мин. до времени получения
- Разница между временными слотами для получения заказа = 30 мин

Пример: Заказ сделан в 16:27

Клиент может выбрать время получения:
17:00 // 17:30 // 18:00 // 18:30 // 19:00 и т.д.





02

Контекстная диаграмма



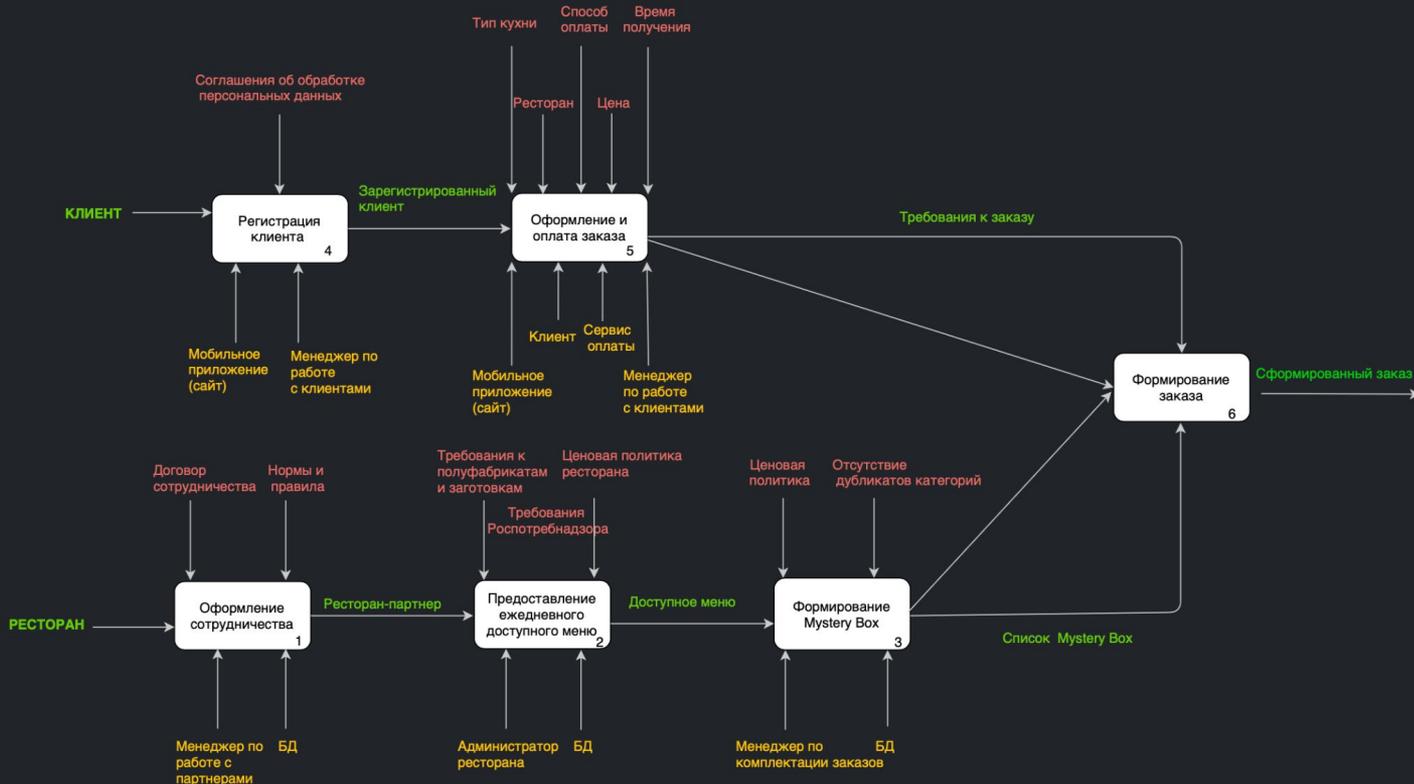


03

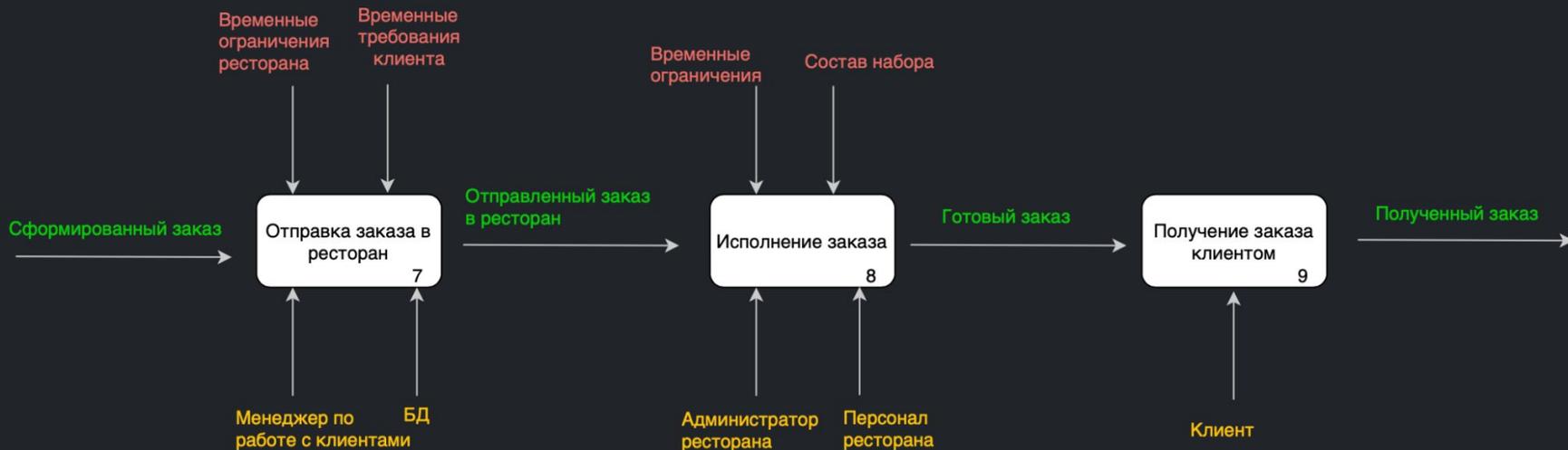
Организационная структура



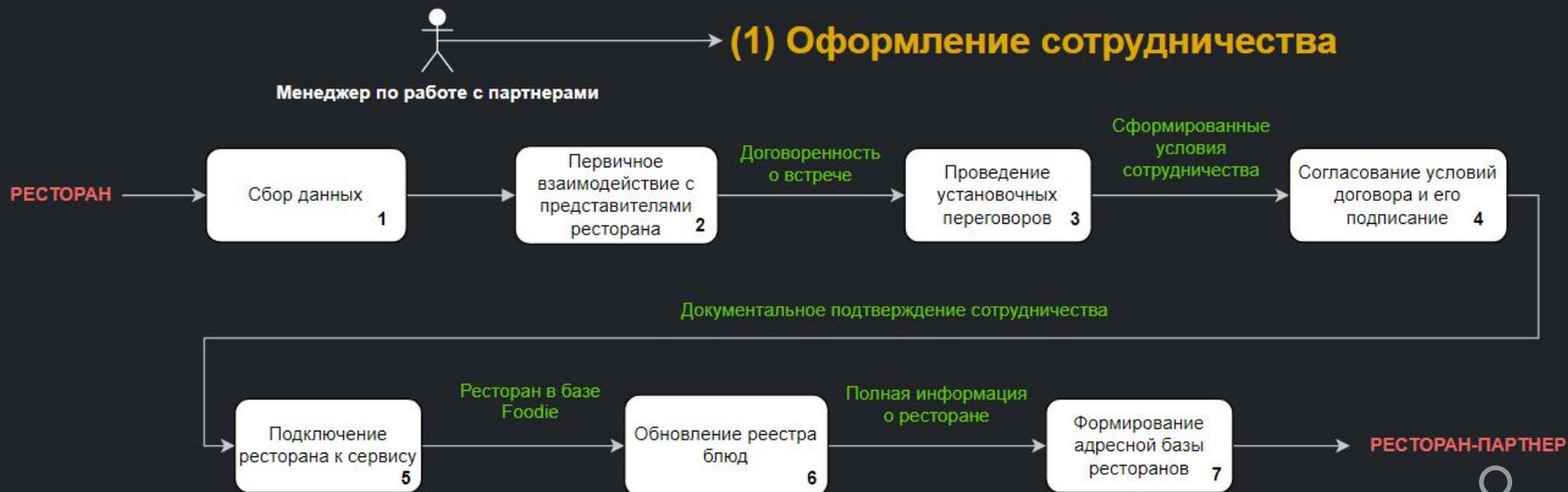
Декомпозиция 1-го уровня



Декомпозиция 1-го уровня

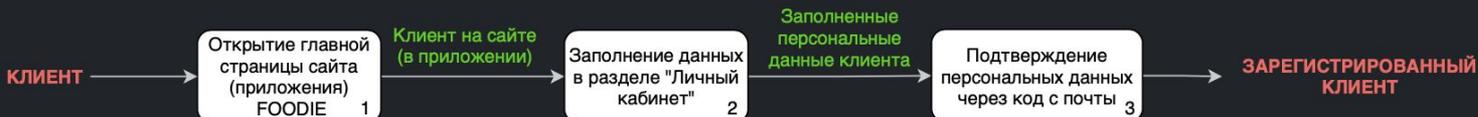
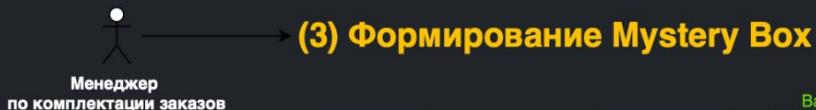


Декомпозиция 2-го уровня





Декомпозиция 2-го уровня

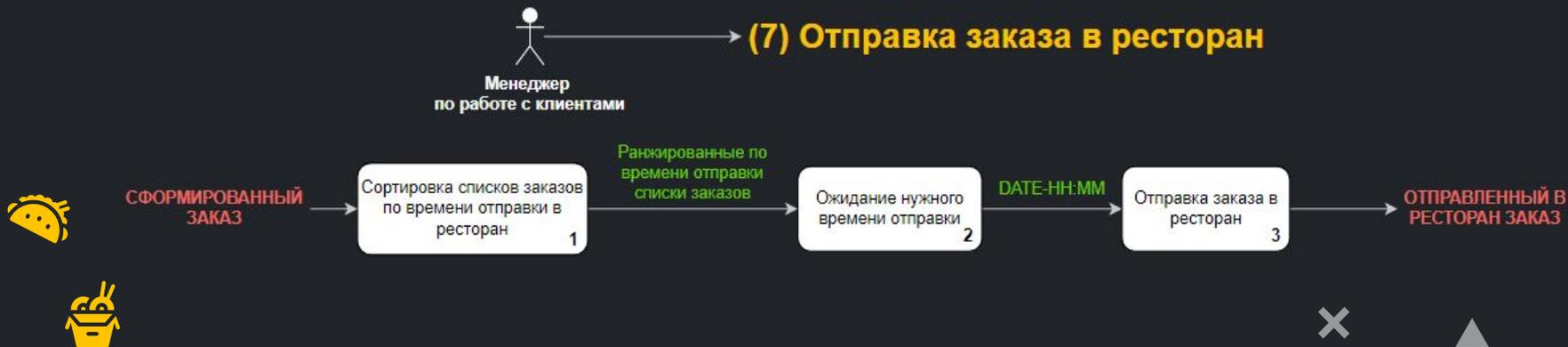




Декомпозиция 2-го уровня

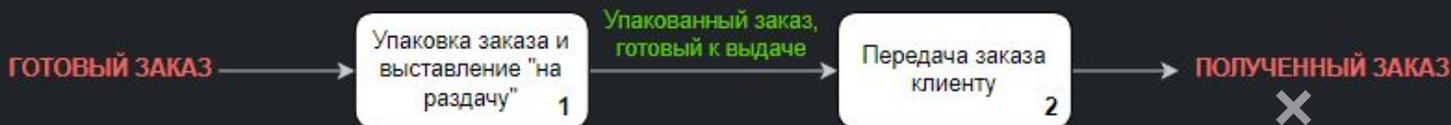
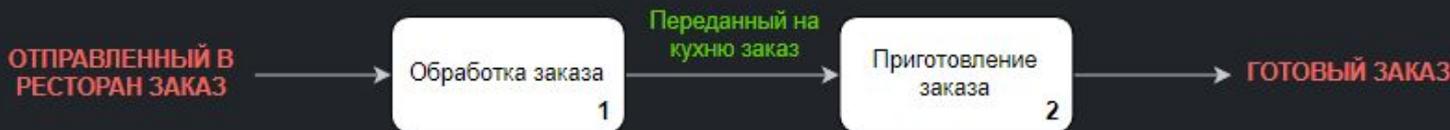


Декомпозиция 2-го уровня





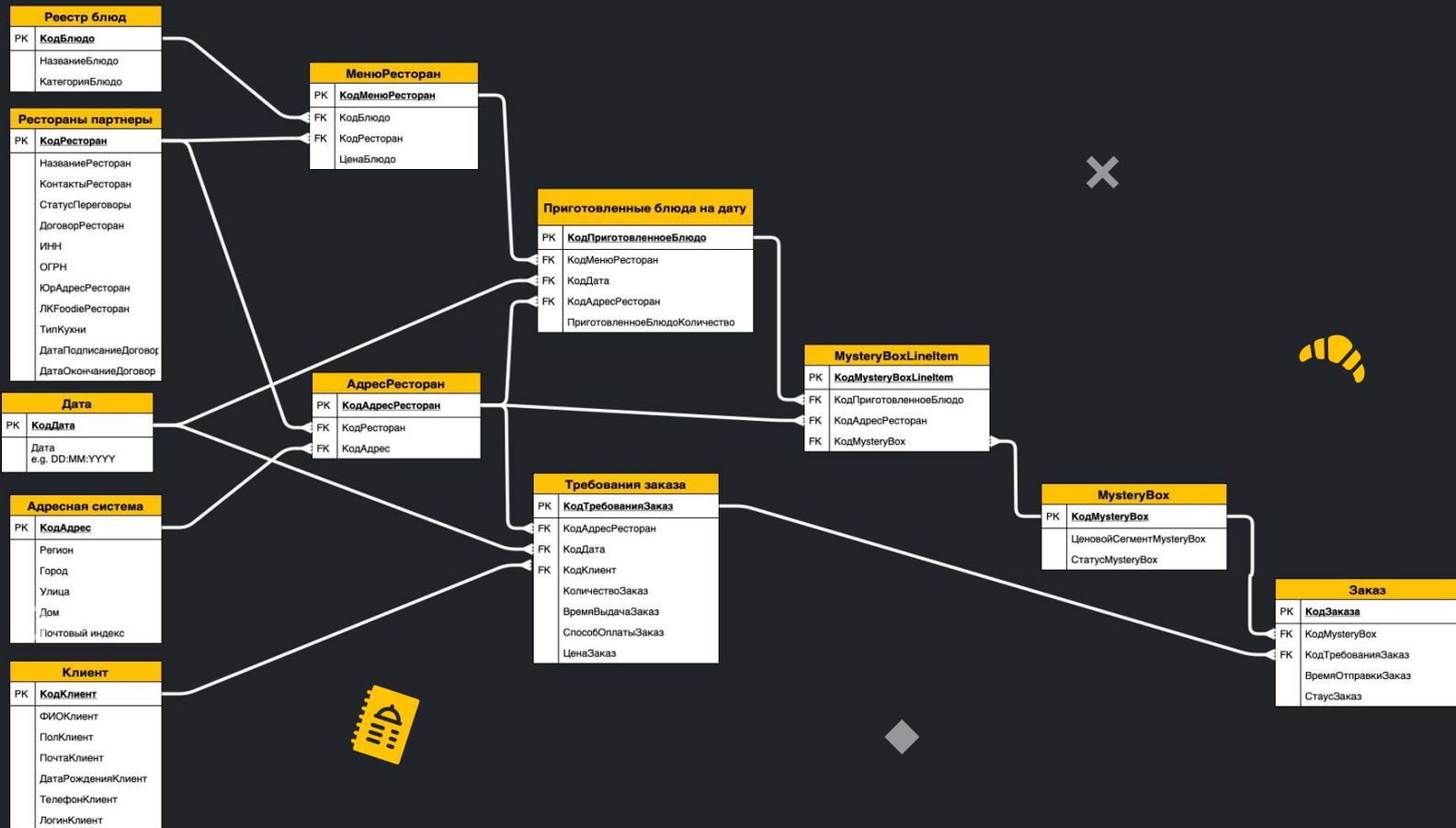
Декомпозиция 2-го уровня





05

Логическая модель данных



06

Поддержка процессов запросами

Бизнес-процесс

Подключение
ресторанов-
партнеров
5 шагов

Внесение
партнером
информации
2 шага

Формирование
Mystery Box
4 шага

Оформление
заказа клиентом
2 шага

Формирование
заказа менеджером
4 шага

Исполнение
заказа
3 шага

Завершение
заказа
1 шаг



Примеры запросов



Этап 1. Подключение ресторанов-партнеров

Внесение нового ресторана партнера

Исполнитель: Менеджер по работе с партнерами

Запрос: C_1_2_PartnerManager

Данные: название ресторана, статус переговоров, номер телефона, тип кухни ресторана

```
INSERT INTO [Рестораны партнеры] (КодРесторан, НазваниеРесторан,  
СтатусПереговоры, КонтактыРесторан, ТипКухни)  
VALUES (:Код, :Название, :СтатусПереговоров, :Контакт, :КухняТип);
```





Примеры запросов



Этап 1. Подключение ресторанов-партнеров

Изменение статуса переговоров

Исполнитель: Менеджер по работе с партнерами

Запрос: U_1_3_PartnerManager

Данные: статус переговоров

```
UPDATE [Рестораны партнеры] SET СтатусПереговоры =:Новый_статус_переговоры  
WHERE НазваниеРесторан=:Название_ресторана;
```





Примеры запросов



Этап 1. Подключение ресторанов-партнеров

Внесение деталей по договору

Исполнитель: Менеджер по работе с партнерами

Запрос: U_1_4_PartnerManager

Данные: указывается новый статус “договор подписан”, вносятся юр. данные

```
UPDATE [Рестораны партнеры] SET СтатусПереговоры =:Новый_статус_переговоры,  
ДоговорРесторан =:НомерДоговора, ИНН=:ИНН, ОГРН=:ОГРН, ЮрАдресРесторан=:  
ЮридическийАдрес, ДатаПодписаниеДоговор=:ДатаПодписанияДоговора,  
ДатаОкончаниеДоговор=:ДатаОкончанияДоговора  
WHERE НазваниеРесторан=:Название_ресторана;
```



Примеры запросов



Этап 1. Подключение ресторанов-партнеров

Внесение деталей по договору

Исполнитель: Менеджер по работе с партнерами

Запрос: U_1_4_PartnerManager

Данные: указывается новый статус “договор подписан”, вносятся юр. данные

```
UPDATE [Рестораны партнеры] SET СтатусПереговоры =:Новый_статус_переговоры,  
ДоговорРесторан =:НомерДоговора, ИНН=:ИНН, ОГРН=:ОГРН, ЮрАдресРесторан=:  
ЮридическийАдрес, ДатаПодписаниеДоговор=:ДатаПодписанияДоговора,  
ДатаОкончаниеДоговор=:ДатаОкончанияДоговора  
WHERE НазваниеРесторан=:Название_ресторана;
```



Примеры запросов



Этап 1. Подключение ресторанов-партнеров

Внесение блюд, которые есть у партнера

Исполнитель: Менеджер по работе с партнерами

Запрос: C_1_6_PartnerManager

Данные: в реестр блюд вносится категория и название блюд всего меню

```
INSERT INTO [Реестр блюд] ( КатегорияБлюдо, НазваниеБлюдо )  
VALUES (:Категория_блюда , :Название_блюда);
```





Примеры запросов



Этап 1. Подключение ресторанов-партнеров

Добавление адресов ресторанов

Исполнитель: Менеджер по работе с партнерами

Запрос: C_1_7_PartnerManager

Данные: каждой точке ресторана ставится адрес

```
INSERT INTO АдресРесторан (КодАдресРесторан, КодРесторан, КодАдрес )  
VALUES (:КодАдресРесторана, :КодРесторана, :КодАдреса);
```





Примеры запросов



Этап 2. Внесение партнером информации

Внесение меню, которое Foodie может использовать

Исполнитель: Администратор ресторана

Запрос: C_2_1_Administrator

Данные: блюда из ресторана с ценами

```
INSERT INTO МенюРесторан (КодМенюРесторан, КодРесторан, КодБлюдо, ЦенаБлюдо)
VALUES (:КодМенюРесторан, :КодРесторан, :КодБлюдо, :ЦенаБлюдо);
```





Примеры запросов



Этап 2. Внесение партнером информации

Внесение блюд на конкретную дату

Исполнитель: Администратор ресторана

Запрос: C_2_2-3_Administrator

Данные: блюда на дату из ресторана с количеством

```
INSERT INTO [Приготовленные блюда на дату] ( КодПриготовленноеБлюдо, КодДата,
КодАдресРесторан, КодМенюРесторан, ПриготовленноеБлюдоКоличество )
VALUES (:КодПриготовленноеБлюдо, :КодДата, :КодАдресРесторана; :
КодМенюРесторана, :ПриготовленноеБлюдоКоличество);
```





Примеры запросов

Этап 3. Формирование Mystery Box



Просмотр доступных блюд от ресторана

Исполнитель: Менеджер по комплектации наборов

Запрос: S_3_1_Former

Данные: выбор ресторана и просмотр, сколько и каких блюд доступно на дату

```
SELECT [Рестораны партнеры].НазваниеРесторан, [Адресная система].Улица, Дата.Дата, [Реестр блюд].  
НазваниеБлюдо, [Приготовленные блюда на дату].ПриготовленноеБлюдоКоличество  
  
FROM [Реестр блюд] INNER JOIN (Дата INNER JOIN ([Адресная система] INNER JOIN ([Рестораны партнеры]  
INNER JOIN (МенюРесторан INNER JOIN (АдресРесторан INNER JOIN [Приготовленные блюда на дату] ON  
АдресРесторан.КодАдресРесторан = [Приготовленные блюда на дату].КодАдресРесторан)  
ON МенюРесторан.КодМенюРесторан = [Приготовленные блюда на дату].КодМенюРесторан) ON  
([Рестораны партнеры].КодРесторан = АдресРесторан.КодРесторан) AND ([Рестораны партнеры].  
КодРесторан = МенюРесторан.КодРесторан)) ON [Адресная система].КодАдрес = АдресРесторан.  
КодАдрес) ON Дата.КодДата = [Приготовленные блюда на дату].КодДата) ON [Реестр блюд].КодБлюдо =  
МенюРесторан.КодБлюдо  
  
WHERE [Рестораны партнеры].НазваниеРесторан=:Название_ресторана AND [Адресная система].  
Улица=:Адрес_улица AND Дата.Дата=CONVERT(DATETIME, :Необходимая_дата);
```



Примеры запросов



Этап 3. Формирование Mystery Box

Формирование Mystery Box

Исполнитель: Менеджер по комплектации наборов

Запрос: C_3_2_Former

Данные: выбор блюд из конкретного ресторана и отнесение к Mystery Box

```
INSERT INTO MysteryBoxLineItem (КодMysteryBoxLineItem, КодАдресРесторан,  
КодMysteryBox, КодПриготовленноеБлюдо )
```

```
VALUES (:КодMysteryBoxLineItem, :КодАдресРесторан, :КодMysteryBox, :  
КодПриготовленноеБлюдо);
```





Критерии полу-автоматического формирования Mystery Box



:Кол-во	#	Ценовой сегмент MB	Условие	Категория блюда = Горячее блюдо ИЛИ Суп	Категория блюда = Салат ИЛИ Закуска ИЛИ ...	Условие для продолжения наполнения	Категория блюда = (Дополнения)
:Требуемое кол-во	1	350	SUM < 300	[Обязательно]: Цена < 200	[Обязательно]: Цена < 100	IF (ЦЕНОВОЙ СЕГМЕНТ - текущая себестоимость набора) > 100 : продолжить процедуру ELSE Сформировать набор	Цена < Дельты предыдущего действия
:Требуемое кол-во	2	550	SUM < 500	[Обязательно]: Цена > 200	[Обязательно]: 100 < Цена < 250		Цена < Дельты предыдущего действия
:Требуемое кол-во	3	850	SUM < 800	[Обязательно]: Цена > 300	[Обязательно]: 150 < Цена < 350		Цена < Дельты предыдущего действия

Использованное блюдо будет исключаться из списка готовых блюд





Примеры запросов



Этап 3. Формирование Mystery Box

Обновление доступных блюд

Исполнитель: Менеджер по комплектации наборов

Запрос: U_3_3_Former

Данные: изменение количества приготовленных блюд на дату

```
UPDATE [Приготовленные блюда на дату] SET [Приготовленные блюда на дату].  
ПриготовленноеБлюдоКоличество =:Новое_количество  
WHERE КодПриготовленноеБлюдо=:Код_приготовленное_блюдо;
```





Примеры запросов



Этап 3. Формирование Mystery Box

Подсчет суммы блюд в каждом Mystery Box

Исполнитель: Менеджер по комплектации наборов

Запрос: S_3_4_Former

Данные: выбор Mystery Box и просмотр “себестоимости”

```
SELECT MysteryBox.КодMysteryBox, Sum(МенюРесторан.ЦенаБлюдо) AS SumOfЦенаБлюдо
FROM МенюРесторан INNER JOIN ([Приготовленные блюда на дату] INNER JOIN (MysteryBox
INNER JOIN MysteryBoxLineItem ON MysteryBox.КодMysteryBox = MysteryBoxLineItem.
КодMysteryBox) ON [Приготовленные блюда на дату].КодПриготовленноеБлюдо =
MysteryBoxLineItem.КодПриготовленноеБлюдо) ON МенюРесторан.КодМенюРесторан =
[Приготовленные блюда на дату].КодМенюРесторан
GROUP BY MysteryBox.КодMysteryBox
HAVING MysteryBox.КодMysteryBox=:Код_Mystery_Box;
```



Примеры запросов



Этап 3. Формирование Mystery Box

Определение стоимости Mystery Box

Исполнитель: Менеджер по комплектации наборов

Запрос: U_3_4_Former

Данные: на основе S_3_4_Former определяется стоимость Mystery Box

```
UPDATE MysteryBox SET MysteryBox.ЦеновойСегментMysteryBox =:Цена_Mystery_Box,  
MysteryBox.СтатусMysteryBox =:Статус_Mystery_Box  
WHERE MysteryBox.КодMysteryBox=:Код_Mystery_Box;
```





Примеры запросов



Этап 4. Оформление заказа клиентом

Регистрация на сайте

Исполнитель: Клиент

Запрос: C_4_2_Client

Данные: введение личных данных

```
INSERT INTO [Клиент] (КодКлиент, ФИОКлиент, ПолКлиент, ПочтаКлиент,  
ДатаРожденияКлиент, ТелефонКлиент, ЛогинКлиент)  
VALUES ( :КодКлиент, :ФИО, :Пол, :Почта, :ДатаРождения, :Телефон, :Логин);
```





Примеры запросов



Этап 4. Оформление заказа клиентом

Оформление заказа

Исполнитель: Клиент

Запрос: C_5_1-6_Client

Данные: указание требований к заказу

```
INSERT INTO [Требования заказа] ( КодТребованияЗаказ, КодДата, КодКлиент,  
КодАдресРесторан, КоличествоЗаказ, ВремяВыдачаЗаказ, СпособОплатаЗаказ,  
ЦенаЗаказ)
```

```
VALUES ( :КодТребованияЗаказ, :КодДата, :КодКлиент, :КодАдресРесторан, :  
КоличествоMysteryBox, CONVERT(TIME, :ВремяВыдачиЗаказа), :СпособОплатыЗаказа,  
:ЦенаMysteryBox);
```





Примеры запросов



Этап 5. Формирование заказа менеджером

Просмотр доступных Mystery Box

Исполнитель: Менеджер по работе с клиентами

Запрос: S_6_1_ClientManager

Данные: просмотр всех Mystery Box на дату в конкретной точке

```
SELECT АдресРесторан.КодАдресРесторан, Дата.Дата, MysteryBox.КодMysteryBox, MysteryBox.ЦеновойСегментMysteryBox
FROM Дата INNER JOIN ((АдресРесторан INNER JOIN (MysteryBox INNER JOIN MysteryBoxLineItem ON MysteryBox.КодMysteryBox =
MysteryBoxLineItem.КодMysteryBox) ON АдресРесторан.КодАдресРесторан = MysteryBoxLineItem.КодАдресРесторан) INNER JOIN
[Приготовленные блюда на дату] ON ([Приготовленные блюда на дату].КодПриготовленноеБлюдо = MysteryBoxLineItem.
КодПриготовленноеБлюдо) AND (АдресРесторан.КодАдресРесторан = [Приготовленные блюда на дату].КодАдресРесторан)) ON Дата.
КодДата = [Приготовленные блюда на дату].КодДата

GROUP BY АдресРесторан.КодАдресРесторан, Дата.Дата, MysteryBox.КодMysteryBox, MysteryBox.ЦеновойСегментMysteryBox
HAVING АдресРесторан.КодАдресРесторан=:Код_адреса_ресторана AND Дата.Дата=CONVERT(DATETIME,;Дата_формирования);
```





Примеры запросов



Этап 5. Формирование заказа менеджером

Формирование заказа

Исполнитель: Менеджер по работе с клиентами

Запрос: С 6_2 _ ClientManager

Данные: выбор Mystery Box с учетом требований клиента, указание время выдачи и внесение статуса заказа

```
INSERT INTO [Заказ] ( КодЗаказ, КодТребованияЗаказ, КодMysteryBox,  
ВремяОтправкиЗаказ, СтатусЗаказ )
```

```
VALUES (:КодЗаказ,:КодТребованияЗаказ, :КодMysteryBox,CONVERT(TIME,:  
ВремяОтправкиЗаказа),:СтатусЗаказ);
```



Критерии **автоматического** формирования времени отправки заказа



Исходные данные	Требования	Условие для обновления
[Требования заказа]. ВремяВыдачиЗаказ	Отправка заказа в ресторан за 30 минут до времени выдачи	IF ([Заказ].ВремяВыдачиЗаказ - [Требования заказа].ВремяОтправкиЗаказ) />= 00:30:00 : [Заказ].ВремяОтправкиЗаказ = [Требования заказа].ВремяВыдачиЗаказ - 00:30:00 ELSE оставить без изменений



Примеры запросов

Этап 5. Формирование заказа менеджером

Изменение статуса доступности Mystery Box

Исполнитель: Менеджер по работе с клиентами

Запрос: U_6_2_ClientManager

Данные: использованный Mystery Box получает статус «не доступен»

```
UPDATE [MysteryBox]
```

```
SET [MysteryBox].СтатусMysteryBox = :НовыйСтатусMysteryBox
```

```
WHERE [MysteryBox].КодMysteryBox =:КодMysteryBox;
```

Примеры запросов

Этап 5. Формирование заказа менеджером

Подтверждение заказа

Исполнитель: Менеджер по работе с клиентами

Запрос: U_order_status_ClientManager

Данные: изменение статуса заказа на «Оплачен»

```
UPDATE [Заказ] SET [Заказ].СтатусЗаказ = :НовыйСтатусЗаказа  
WHERE [Заказ].КодЗаказ=:КодЗаказа;
```



Примеры запросов



Этап 6. Исполнение заказа

Выбор заказов для отправки по времени

Исполнитель: Менеджер по работе с клиентами

Запрос: S_6_2_ClientManager

Данные: выбор времени и список Mystery Box

```
SELECT Заказ.ВремяОтправкиЗаказ, Дата.Дата, Заказ.КодЗаказ
FROM (Дата INNER JOIN [Требования заказа] ON Дата.КодДата = [Требования заказа].
КодДата) INNER JOIN Заказ ON [Требования заказа].КодТребованияЗаказ = Заказ.
КодТребованияЗаказ
WHERE Заказ.ВремяОтправкиЗаказ=CONVERT(DATETIME,:Время_отправки_заказа)
AND Дата.Дата=CONVERT (DATETIME, :Дата_заказа);
```



Примеры запросов



Этап 6. Исполнение заказа

Отправка заказа в ресторан

Исполнитель: Менеджер по работе с клиентами

Запрос: U_order_status_ClientManager

Данные: изменение статуса заказа на «Отправлен в ресторан»

```
UPDATE [Заказ] SET [Заказ].СтатусЗаказ = :НовыйСтатусЗаказа  
WHERE [Заказ].КодЗаказ=:КодЗаказа;
```





Примеры запросов



Этап 6. Исполнение заказа

Отправка заказа на кухню

Исполнитель: Администратор ресторана

Запрос: U_8/9_Administrator

Данные: изменение статуса заказа на «Готовится»

```
UPDATE Заказ SET Заказ.СтатусЗаказ =:Новый_статус_заказа  
WHERE Заказ.КодЗаказ=:Код_заказа;
```





Примеры запросов

Этап 6. Исполнение заказа



Выдача заказа

Исполнитель: Администратор ресторана

Запрос: U_8/9_Administrator

Данные: изменение статуса заказа на «Готов к выдаче»

```
UPDATE Заказ SET Заказ.СтатусЗаказ =:Новый_статус_заказа  
WHERE Заказ.КодЗаказ=:Код_заказа;
```





Примеры запросов

Этап 7. Завершение заказа



Завершение заказа

Исполнитель: Администратор ресторана

Запрос: U_8/9_Administrator

Данные: изменение статуса заказа на «заказ завершен»

```
UPDATE Заказ SET Заказ.СтатусЗаказ =:Новый_статус_заказа  
WHERE Заказ.КодЗаказ=:Код_заказа;
```





Примеры запросов

Дополнительные функции



Проверка срока действия договора с партнерами

Исполнитель: Менеджер по работе с партнерами

Запрос: S_contract_time_PartnerManager

Данные: выбор даты начала и окончания договора с рестораном

```
SELECT [Рестораны партнеры].НазваниеРесторан, [Рестораны партнеры].  
ДоговорРесторан, [Рестораны партнеры].ДатаОкончаниеДоговор  
FROM [Рестораны партнеры]  
WHERE НазваниеРесторан=Название_Ресторана;
```





Примеры запросов

Дополнительные функции



Удаление блюда из меню, которое предоставил ресторан

Исполнитель: Администратор ресторана

Запрос: D_dish_delete _ Administrator

```
DELETE FROM [МенюРесторан] WHERE КодМенюРесторан=:КодМенюРесторан
```





Примеры запросов

Дополнительные функции



Изменение времени выдачи заказа клиентом

Исполнитель: Клиент

Запрос: U_order_time_Client

```
UPDATE [Требования заказа] SET [Требования заказа].ВремяВыдачаЗаказ =  
CONVERT(TIME ,:НовоеВремяВыдачиЗаказа)
```

```
FROM [Требования заказа]
```

```
JOIN [Заказ] ON [Требования заказа].КодТребованияЗаказ = [Заказ].  
КодТребованияЗаказ
```

```
WHERE [Заказ].КодЗаказ=:Код_заказа;
```



Примеры запросов

Дополнительные функции



Изменение времени отправки заказа менеджером

Исполнитель: Менеджер по работе с клиентами

Запрос: U_order_time_ClientManager

```
UPDATE [Заказ]
```

```
SET [Заказ].ВремяОтправкаЗаказ =:НовоеВремяОтправкиЗаказа
```

```
WHERE [Заказ].КодЗаказ =:КодЗаказа;
```





Примеры запросов

Дополнительные функции



Проверка статуса заказа

Исполнитель: Клиент

Запрос: S_order_status_Client

```
SELECT [Заказ].КодЗаказ, [Клиент].ЛогинКлиент, [Рестораны партнеры].НазваниеРесторан, [Адресная система].Улица, [Заказ].СтатусЗаказ
FROM [Адресная система]

JOIN ([Рестораны партнеры] INNER JOIN ([АдресРесторан INNER JOIN (МенюРесторан INNER JOIN [Приготовленные блюда на дату] ON МенюРесторан.
КодМенюРесторан = [Приготовленные блюда на дату].КодМенюРесторан) ON АдресРесторан.КодАдресРесторан = [Приготовленные блюда на дату].
КодАдресРесторан) INNER JOIN (Клиент INNER JOIN ([Требования заказа] INNER JOIN Заказ ON [Требования заказа].КодТребованияЗаказ = Заказ.
КодТребованияЗаказ) ON Клиент.КодКлиент = [Требования заказа].КодКлиент) ON АдресРесторан.КодАдресРесторан = [Требования заказа].
КодАдресРесторан) ON ([Рестораны партнеры].КодРесторан = АдресРесторан.КодРесторан) AND ([Рестораны партнеры].КодРесторан =
МенюРесторан.КодРесторан)) ON [Адресная система].КодАдрес = АдресРесторан.КодАдрес

GROUP BY Заказ.КодЗаказ, Клиент.ЛогинКлиент, [Рестораны партнеры].НазваниеРесторан, [Адресная система].Улица, Заказ.СтатусЗаказ

HAVING ((([Заказ].КодЗаказ)=:Код_заказа));
```



Примеры запросов

Дополнительные функции



Анализ числа заказов у клиентов

Исполнитель: Менеджер по работе с клиентами

Запрос: S_order_quantity_ClientManager

```
SELECT Клиент.КодКлиент, Клиент.ФИОКлиент, Count(Заказ.КодЗаказ) AS  
CountOfКодЗаказ
```

```
FROM (Клиент INNER JOIN [Требования заказа] ON Клиент.КодКлиент = [Требования  
заказа].КодКлиент) INNER JOIN Заказ ON [Требования заказа].КодТребованияЗаказ =  
Заказ.КодТребованияЗаказ
```

```
GROUP BY Клиент.КодКлиент, Клиент.ФИОКлиент;
```





Спасибо!

