***Приложение"Электронное меню" для сети ресторанов"Гурман"***

**1 Business purpose / Бизнес-назначение**

Данное приложение позволит сократить расходы в системе обслуживания клиентов, также оно будет вести учёт заказов.

**2 Business scope / Бизнес-рамки**

С помощью данного приложения официант будет делать заказ, заказ отправляется на панель заказов, которая находится на кухне. Второй официант будет разносить заказы. Без этого проекта официанты принимают, передают, разносят заказы.

**3 Business overview / Обзор бизнеса**

**Рестораны и бары по уровню обслуживания и номенклатуре предоставляемых услуг подразделяют на три класса — «люкс», «высший» и «первый», которые должны соответствовать следующим требованиям:**

* **«люкс» — широкий выбор услуг, предоставляемых потребителям, высокий уровень комфортности и удобство размещения потребителей в зале, широкий ассортимент оригинальных, изысканных заказных и фирменных блюд, изделий, характерных для ресторанов, широкий выбор заказных и фирменных напитков, коктейлей для баров, изысканная сервировка столов, фирменный стиль, специфика подачи блюд, эксклюзивность и роскошь интерьера;**
* **«высший» — большой выбор услуг, предоставляемых потребителям, комфортность и удобство размещения потребителей в зале, разнообразный ассортимент оригинальных, изысканных заказных и фирменных блюд и изделий для ресторанов, широкий выбор фирменных и заказных напитков и коктейлей — для баров, фирменный стиль, изысканность и оригинальность интерьера;**
* **«первый» — определенный выбор услуг, предоставляемых потребителям, разнообразный ассортимент фирменных блюд и изделий и напитков сложного изготовления, характерный для ресторанов, широкий или специализированный ассортимент напитков и коктейлей, в том числе заказных и фирменных для баров, гармоничность и комфортность интерьера.**

Наш ресторан относится к ресторанам "среднего класса".

По ГОСТ ресторан "**первого**" класса:

Первому классу соответствует гармоничность, комфортность и выбор услуг, широкий ассортимент фирменных блюд и изделий, а также напитков сложного приготовления для ресторанов, набор напитков, коктейлей несложного приготовления – для баров.

**4 Stakeholders / Заинтересованые лица**

1. Владелец ресторана(получение прибыли, спустя определенное время)
2. Шеф-повар
3. Повара
4. Бухгалтер
5. Официанты(сокращение с 4 до 2, перераспределение работы с внедрениям проекта)
6. Уборщики
7. Охранники

Шеф-повар и повара будут сразу получать заказ, не придётся ожидать пока официант его передаст, 25% ускорение доставки заказа клиенту. Следовательно клиенты будут обслуживаться быстрее.

**5 Business environment / Организационная среда**

Было рассмотрено два варианта проекта: клиент может заказывать с помощью планшета(более дорогостоящий вариант) и заказы делает официант с помощью планшета(более дешёвый).

У нас не богатый бюджет, наш бизнес не может себе позволить вкладывать очень большую сумму куда либо, в том числе и в этот проект, и мы заинтересованы в как можно более быстром способе получения прибыли. Поэтому был выбран второй вариант.

**6 Goal and Objective / Цели и результаты**

Цели: снижение затрат и реорганизацию передачи заказов.

**7 Business model / Бизнес-модель**

Будет закуплено 2 планшета, 1 настенный широкоэкранный телевизор, сервер.

На планшеты будет установлено приложение, с помощью которого можно будет отправлять на сервер заказы и менять состояния заказа(В процессе, Сделан, Не оплачен)(Подробнее в 15-17 пунктах). Сервер будет выводить на телевизор информацию.

Клиент будет заказывать блюда с помощью планшета, разносчик будет редактировать состояния заказов.

**8 Information environment / Информационная среда**

Для достижения данной цели достаточно одного(этого) проекта.

В дальнейшем ,когда увеличится прибыль и проект окупиться, можно будет реализовать первый вариант проекта, в результате которого можно будет оставить только одного официанта. Также перейдя к этому проекту мы минимизируем затраты при изменение меню.

**9 Business processes / Бизнес-процессы**

Смотри User case.vsd, Active diagram.vsd, Sequence diagram.vsd

**10 Business operational policies and rules / Политики и правила**

Правила для

1. Приёмщик

* Только приёмщик может создавать заказы, отменять, переводить только в состояние "Оплачен"

1. Разносчик

* Может только изменить состояние заказа на "Доставлен" или "Не доставлен"

1. Повар

* Может переводить состояние заказа только в "В процессе" или "Готов"

Три выше описанных пользователя имеют доступ только к таблице неоплаченных заказов.

1. Бухгалтер

* Не имеет доступа к неоплаченным заказам, только к оплаченным, которые помещаются в отдельную БД

1. Администратор

* Имеет те же правила что и бухгалтер
* Только он может редактировать меню

**11 Business operational constraints / Ограничения деятельности**

Процессы в системе могут происходить параллельно, поэтому требуется многопоточность и следовательно требуется как-то решить проблемы многопоточности(одновременное изменение данных, например). Также они должны выполняться достаточно быстро(1 обновления в секунду будет достаточно.

**12 Business operation modes / Режимы работы**

В случае неполадок с сервером или интернетом, официантам и поварам придётся вернуться к старому режиму работы(как до внедрения проекта). Данные о заказах, назначенных блюдах будут сохранены в приложениях на планшетах, когда сервер будет починен, приложение нужно будет перезапустить.

**13 Business operational quality / Качество выполнения деятельности**

Качество системы будет измеряться в ***кол-ве обслуживаемых клиентов***, и сравнении этого показателя с показателям до внедрения системы, если показатели примерно равны или кол-во обслуживаемых клиентов больше допроектного значения, то значит проект успешен.

На выше описанный выше показатель будет влиять удобство использования и скорость работы(добавления заказов, изменения их состояний) официантов. Если ***скорость работы официантов*** выше или примерно равна допроектному показателю(с учётом того, что официантов меньше), то значит проект успешен.

**14 Business structure / Организационная структура**

1. Хозяин ресторана

В системе будет иметь роль администратора. Он будет получать прибыль от проекта.

1. Повара

Получают заказы от официантов, готовят блюда, распределяют между собой блюда. Данная система позволит им организовать распределение блюд между собой и в более наглядном виде иметь информацию о порученным им заказах.

1. Официанты

Принимаю и доставляют заказы, также принимают деньги. Позволит разделить на две узко направленных должности: приёмщика и разносчика. Так же система позволит избежать ошибок при приёме заказов.

**15 User requirements / Требования пользователей**

***Функциональные*** требования **повара** к системе

* Можно назначать заказ на конкретного повара(на себя или другого)
* Можно отмечать готовые заказы

***Нефункциональные*** требования **повара** к системе

* Скорость обновления списка заказов
  + Кол-во обновлении в секунду(1 обновление в секунду)
* Удобность использования
  + Быстрый поиск заказа(Заказы отсортированы)
  + Внешний вид приятен для глаз.
  + Интерфейс понятен, прост в использовании.
* Отказоустойчивость
  + Сохранность данных
  + Корректное состояние системы при ошибках, возможность отката к допустимому состоянию

***Функциональные*** требования **официанта** к системе

* Можно создавать и просматривать заказы у конкретного столика
* Можно удалять заказы
* Можно завершить, отменить заказ

***Нефункциональные*** требования **официанта(приёмщика и разносчика)** к системе

* Скорость обновления списка заказов
  + Кол-во обновлении в секунду(1 обновление в секунду)
* Удобность использования
  + Быстрый поиск заказа(Заказы отсортированы)
  + Внешний вид приятен для глаз.
  + Интерфейс понятен, прост в использовании.
* Отказоустойчивость
  + Сохранность данных
  + Корректное состояние системы при ошибках, возможность отката к допустимому состоянию

***Функциональные*** требования **бухгалтера** к системе

* Можно просматривать сохранять данные в виде excel-файла
* Можно сравнить доходы из одного временного промежутка с другим

***Нефункциональные*** требования **бухгалтера** к системе

* Удобность использования
  + Быстрый поиск заказов(Заказы отсортированы)
  + Внешний вид приятен для глаз.
  + Интерфейс понятен, прост в использовании.
* Отказоустойчивость
  + Сохранность данных
  + Корректное состояние системы при ошибках, возможность отката к допустимому состоянию

***Функциональные*** требования **Администратора** к системе

* Можно просматривать сохранять данные в виде excel-файла
* Можно добавлять и изменять старые виды блюд и блюда соответствующего типа

***Нефункциональные*** требования **Администратора** к системе

* Удобность использования
  + Быстрый поиск заказов, блюд и их видов(Заказы отсортированы)
  + Внешний вид приятен для глаз.
  + Интерфейс понятен, прост в использовании.
* Отказоустойчивость
  + Сохранность данных
  + Корректное состояние системы при ошибках, возможность отката к допустимому состоянию

**16 Operational concept / Концепция использования системы**

На планшеты будет установлено приложение.

С его помощью можно просматривать кто что заказал, **отменять** и создавать заказы. У официантов будет по одному планшету**,** этот планшет будет использоваться для сброса заказов у отдельного столика(сбрасывать будет приёмщик)

Один официант будет выполнять роль приёмщика заказов, второй разносчика. Первый только ходит между столиками, принимает заказы и деньги, второй разносит заказы. Повара будут брать заказы, отмечать те которые они будут делать, и отмечать готовые.

Приложение будет таким образом разработано, что можно будет с помощью специальной программы редактировать меню, программа будет достаточно простой, чтобы можно было отредактировать без участия программистов.

**Поддержка системы:** раз в месяц будут приглашаться программисты создавшие систему для проверки работы системы

**17 Operational scenarios / Сценарии эксплуатации**

Официант просматривает меню выбирает нужные клиенту блюда, нажимает на кнопку заказать. Далее выводится список заказанных блюд и спрашивается: вы уверены что хотите заказать именно эти блюда?

Будет предложено три варианта:

* Да. После нажатия заказ отправляется на сервер и содержимое выводится на экран, который находится на кухне.
* Вернуться. Список заказанных блюд будет сохранён и клиент может дозаказать или отменить некоторые блюда.
* Сброс. Список заказанных блюд будет сброшен, клиент должен будет сформировать список заново.

Созданный заказ имеет состояние "Не готовится", повар выбирает заказ, который не готовится и переводит его состояние в "В процессе", когда заказ будет готов в состояние "Готов". Когда заказ в состоянии готов, разносчик доставляет заказ и переводит в состояние "Доставлен". Когда клиент оплатит заказ, то приёмщик удалит заказ.

Бухгалтер может выбрать временной промежуток и сохранять таблицу отчёт в виде таблицы.

**18 Project constraints / Проектные ограничения**

Зарплата приёмщикам заказов: N приёмщиков \* зарплата

Обслуживание сервера: N1 рублей

Обслуживание планшетов: 3 планшета \* стоимость обслуживания

Обслуживание телевизора: N2 рублей

Ежемесячная проверка оборудования: N3 рублей

Стоимость проекта:

* стоимость планшета \* 3 планшета
* стоимость телевизора
* стоимость сервера
* оплата труда программистов

Время на выполнение проекта: 2 недели создание приложение, 1 неделя на установку, настройку оборудования, тестирование в реальных условиях

Время через которое проект окупиться: Стоимость проекта / Зарплата приёмщиков заказов + (Зарплата приёмщиков заказов + 1) / (обслуживание планшетов ,сервера , телевизора + ежемесячная проверка оборудования)