# Терминология:

**ДГ** – аппаратно-программный комплекс, призванный для улучшения качества обслуживания в заведениях ресторанного типа.

**Клиент** – человек заказавший систему.

**Заведение** – непосредственно набор помещений, в которые установлено оборудование ДГ, данные залы объединяет наличие единого центра (сервера) обработки данных со столов и с приборов официантов. Заведение может состоять из нескольких залов.

**Root** – роль доступа к ПО ДГ позволяющая делать всёёёёё, основной задачей данной роли будет обновление ПО, создание резервных копий, и восстановление.

**Engineer** – роль доступа к ПО ДГ предназначенная для заведений новых клиентов и помощи им и их администраторам в настройки ПО.

**Director** – роль доступа к ПО направленная на просмотр отчетности по своим заведениям, и имеющая права по изменению списка и ролей аккаунтов имеющих доступ до своих заведений.

**Admin** – роль доступа к ПО ДГ предоставляющая доступ к настройке данных заведений, конфигурации залови предустановленных расписаний, и удалению записей о сотрудниках из базы в заведениях доступ до которых ему дал **Director**.

**Manager** – роль доступа к ПО ДГ позволяющая задавать расписания работы, добавлять/увольнять, мониторить активность и возможно имеющий доступ до некоторых отчетов по сотрудникам подотчетного ему заведения

**Hostes** – роль доступа к ПО ДГ позволяющая открывать/закрывать рабочий день (смену), следить за активностью и проводить диагностику системы.

# Программное обеспечение для системы ДГ должно будет обеспечить.

* Во-первых система должна быть многопользовательской с ограничением доступа до данных и функционала по определенным ролям доступа (Root –).
* Стабильную работу системы (в случае детектирования отказа частей системы должны быть предусмотрены способы оповещения требуемых людей с описанием всех инструкций, что нужно сделать, чтобы устранить неисправность).
* Максимальную эффективность периферийного оборудования (для некоторой функциональности видимо придется поддерживать постоянное соединение между определенными частями системы)
* Предоставлять удобный и понятный интерфейс для работы с системой (здесь имеются ввиду следующие группы сотрудников: хостесы и менеджеры, т.е. интерфейс для задания расписаний и контроля работы официантов должен быть максимально прост и удобен). Но при этом должна оставаться возможность кастомизировать любой параметр системы (один из вариантов - это переключение между простым и продвинутым видом интерфейса)
* Система должна предоставлять краткий набор отчетов с возможность фильтрации и сортировки по любому из параметров отчета. Также должен быть предоставлен инструментарий для формирования произвольных отчетов за произвольный период.
* Время, которое хранятся данные для отчетов, должно настраиваться персонально для каждого заведения, при этом величина времени хранения данных для отчетов и объем данных для отчетов не должен влиять на быстродействие и работоспособность системы в целом и тем более в других заведениях, учетные записи которых могут находиться на этом же сервере.
* ПО должно предоставить возможность удаленного открытия и закрытия столов (вероятнее всего данная функциональность будет привязана к аппаратно-программному комплексу R-Keeper).
* Предоставлять удобные средства диагностики системы, не требующие присутствия специалиста на месте.
* Иметь инструментарий симуляции активности (для проведения тестирования системы в отсутствии оборудования)
* Требуется наличие возможности логирования активности с возможностью воспроизведения записанной активности с целью воспроизведения ситуаций в которых происходили проблемы.
* Интерфейс и функциональность ПО для роли менеджера и хостеса должна быть ориентирована на работе на планшетных компьютерах.
* Также требуется предусмотреть возможность переноса сервера на новое место без остановки работы обслуживаемых заведений на долгое время.

# Карта сайта.

Карта сайта будет зависеть от роли аккаунта, под которым авторизовался пользователь. Причем часть страниц доступная Hostess – будет недоступна Director’у, такой подход облегчит отладку функционала и позволит не перегружать функциональностью главенствующие роли.

## Карта сайта для пользователя с правами Root.

Карту сайта для пользователя с правами root – описать на данный момент достаточно сложно – потому что скорее всего будет админка того портала на базе которого портала или CMS на базе которого все это будет сделано. Сейчас в качестве основы предполагается брать уже готовое ядро, скорее всего это будет Drupal последней версии, теоретически при нагрузке в 500000 простых запросов в сутки он должен будет справится (это приблизительно 5 запросов в секунду), при увеличении количества запросов целесообразно будет подумать о переходе с протокола http на постоянное соединение (для этого скорее всего придется отказаться от PHP и перейти на java), написания собственного клиентского приложения для хостесов и администраторов, вынесение статистики на отдельный сервер, или просто перенос части заведений на другой сервер.

По большому счету функциональность сервера для пользователя с правами root должна заключаться странице авторизации и странице с тремя кнопками,

* Закрыть сервер на техническое обслуживание
* Обновить данные
* Открыть сервер

## Карта сайта для пользователя с правами Engineer.

* Список клиентов.
  + Список сотрудников клиента имеющих доступ к системе (администраторы, менеджеры, хостесы)
  + Список общих отчетов
  + Список заведений
    - Список сотрудников заведения
    - Список столов в заведении
      * Диагностика работы передатчиков
    - Список приемников
    - Список залов в заведении
      * Список схем расположений столов
        + Список разбиений схемы по зонам
    - Список шаблонов расписаний
    - Список шаблонов сообщений

## Карта сайта для пользователя с правами Director

* + Список сотрудников клиента имеющих доступ к системе (администраторы, менеджеры, хостесы)
  + Список общих отчетов
  + Список заведений
    - Список сотрудников заведения
    - Список отчетов по заведению
    - Страница мониторинга текущего рабочего дня
    - Мониторинг текущей смены.

## Карта сайта для пользователя с правами Manager

* Список сотрудников заведения
* Диагностика столов
* Диагностика приемников сообщений
* Календарь
  + Добавление/редактирование рабочего дня
  + Расширенные настройки добавления/редактирования рабочего дня
  + Мониторинг текущей активности
    - Посмотреть статистику по определенному официанту в рамках текущей смены
    - Отправить сообщение
    - Открытие/закрытие заведения
    - Страница со списком забронированных столиков
    - Получить сообщение (отправление сообщения будет предусмотрено с бара)
    - Операции текущей сменны
      * Бронирование столика
      * Снятие брони
      * Открытие столика
      * Закрытие столика
      * Перенос столика
      * Объединение столов
      * Разъединение столов
      * Смена официанта
      * Назначение дополнительного официанта
      * Открытие смены
      * Закрытие смены
      * Просмотр текущей информации по столику

## Карта сайта для пользователя с правами Hostel

* Диагностика столов
* Диагностика приемников сообщений
  + Мониторинг текущей активности
    - Посмотреть статистику по определенному официанту в рамках текущей смены
    - Отправить сообщение
    - Открытие/закрытие заведения
    - Страница со списком забронированных столиков
    - Получить сообщение (отправление сообщения будет предусмотрено с бара)
    - Операции текущей сменны
      * Бронирование столика
      * Снятие брони
      * Открытие столика
      * Закрытие столика
      * Перенос столика
      * Объединение столов
      * Разъединение столов
      * Смена официанта
      * Назначение дополнительного официанта
      * Открытие смены
      * Закрытие смены
      * Просмотр текущей информации по столику

# Обоснования выбранных технологий

Как упоминалось выше, первоначальный вариант системы предлагается запустить на базе открытой CMS Drupal. На первоначальном этапе работы PHP 5 + Drupal + mysql должны будут справится с объемом работ. Данное предположение выведено из того что:

* каждое новое заведение потребует заведения 10-20 дополнительных таблиц (mysql должен позволить работать с базой в 1000 таблиц).
* Предполагая что в среднем в заведении активно 20 столов к которым требуется подходить раз в пять минут, и заведение работает 12 часов в день – то таблицы событий в заведении будет увеличиваться на 1000 – 1500 записей в день (по личному опыту – линейные таблицы, без дополнительных связей и без дополнительных текстовых и бинарных полей могут достигать 500 тыс. без сильного ущерба быстродействию системе)
* По утверждениям тестов системы Drupal – она может выдержать 500 тыс. запросов в сутки. Что при группировке данных нескольких событий в один запрос и отсылки такого запроса раз в полминуты (за исключением специализированных запросов) с лихвой перекрывает требуемую производительность.
* В Drupal в данный момент я раздираюсь лучше всего. Да и она является одной из самый популярных CMS на PHP

# Требования к разработчику.

* Знание PHP 5 + знание по объектно-ориентированному программированию.
* Хорошее знание JavaScript + JQuery. Потому что внешняя морда предполагается кросбраузерной + jquery хорошо поддерживается в Drupal.
* Drupal – само собой, хотя из Drupal достаточно знать только работу с темами – все остальное либо уже сделано, либо будет прикручено стандартными средствами.
* Желательно наличия умения работать с системой контроля версий Git + Gerrit.

# Ограничения и допущения разрабатываемой системы

* На данный момент предполагается, что стол не может переноситься из одного зала в другой. Данное ограничение сделано чтобы не вносить в логику создания расписания – проверку что два зала не пересекаются по столам, да и по тому что не смог вспомнить хотя бы одно заведение в котором это бы практиковалось, как правило в заведении каждый зал имеет свой интерьер и поэтому столы просто нельзя переносить.
* В системе будет предусмотрен вариант того что один стол/зону могут обслуживать несколько официантов (логика кому отсылать в таком случае оповещения – либо будет настраиваемой, либо требует обсуждения. Из вариантов логики могу предложить:
  1. Отсылать сообщения всем официантам
  2. Отсылать только одному – самому свободному
  3. Отсылать тому, кто подходил последний раз.
  4. Отсылать только тому кто открывал стол.
  5. Отсылать по очереди

).

Оглавление

[1. Терминология: 1](#_Toc346580821)

[2. Программное обеспечение для системы ДГ должно будет обеспечить. 1](#_Toc346580822)

[3. Карта сайта. 2](#_Toc346580823)

[3.1. Карта сайта для пользователя с правами Root. 2](#_Toc346580824)

[3.2. Карта сайта для пользователя с правами Engineer. 3](#_Toc346580825)

[3.3. Карта сайта для пользователя с правами Director 3](#_Toc346580826)

[3.4. Карта сайта для пользователя с правами Manager 3](#_Toc346580827)

[3.5. Карта сайта для пользователя с правами Hostel 4](#_Toc346580828)

[4. Обоснования выбранных технологий 4](#_Toc346580829)

[5. Требования к разработчику. 5](#_Toc346580830)

[6. Ограничения и допущения разрабатываемой системы 5](#_Toc346580831)