Тестовое

Техническое задание на API "Аренда квартир".

1. Общие сведения
   1. Наименование системы
      1. Полное наименование системы

API аренды квартир.

* + 1. Краткое наименование системы

АР, системa.

* 1. Основания для проведения работ

Работа выполняется на основе тестового задания от Колосова Александра Андреевича

* 1. Наименование организаций – Заказчика и Разработчика
     1. Заказчик

Заказчик: ОАО Заказчик

Адрес: Екатеринбург …

Телефон: +7 343 111 111 1

* + 1. fphf,Разработчик

Разработчик: ОАО Разработчик

Адрес: Екатеринбург …

Телефон: +7 343 222 222 2

* 1. Плановые сроки начала и окончания работы
  2. Порядок оформления и предъявления заказчику результатов работ

Работы по созданию сдаются Разработчиком поэтапно в соответствии с календарным планом проекта.

1. Назначение и цели создания системы
   1. Назначение системы

АР предназначена для упрощения процесса поиска квартиры в аренду.

* 1. Цели системы

АР создается с целью:

* Создания единой системы поиска квартиры в аренду;
* Повышения скорости оформления договора об аренде.

1. Характеристика объектов автоматизации
   1. Общие сведения

Данное ТЗ описывает принципы функционирования сервиса по поиску аренды квартир на различных условиях, условия взаимодействия пользователя с базой данных сервиса, а также возможности взаимодействия с другими пользователями сервиса. Основной задачей сервиса является предоставление пользователю информации о квартирах, сдающихся в аренду, выбор основных критериев жилищных условий, выбор сроков проживания и расчет стоимости и формирование договора аренды.

* 1. Основные термины и понятия

Квартира может сдаваться в аренду на срок:

* длительное проживание (больше месяца);
* краткосрочная аренда ( от суток до одного месяца);
* проведение бизнес встреч / сессий настольных игр и т.п. ( почасовая оплата );

Квартира имеет следующие характеристики:

* Кличка;
* класс комфорта (ремонта);
* год постройки дома;
* количество спален;
* количество гостиных;
* количество сан узлов;
* количество кладовок и подсобных помещений;
* общая площадь;
* Описание (стиль ремонта, информация о соседях и прочее);
* Ставка почасовой аренды;
* Стоимость.

1. Требования к системе
   * 1. Требования к структуре и функционированию системы

Информация в системе должна быть закодирована (зашифрована).

Требования для системы шифрования:

- Все сущности должны быть идентифицированы;

- Для зашифрования и расшифрования информации необходимы ключи;

* При изменении ключевой информации зашифрованное сообщение должно быть существенно изменено и не может быть расшифровано предыдущим ключом;
* При изменении исходной информации, зашифрованное сообщение должно существенно изменено при использовании одного и того же ключа;
* Знание алгоритма шифрования не позволяет расшифровать исходную информацию;
* Передача зашифрованных сообщений по незащищенным каналам связи не должна наносить ущерб информации и позволять расшифровать сообщение;
* Метод шифрования должен быть криптостойким;

Система должна быть централизованной, т.е. все данные по квартирам должны располагаться в одной базе данных.

Требования к получению каталога квартир:

* Каталог должен иметь возможность упорядочивания по имеющимся характеристикам;
* Каталог должен иметь возможность сортировки по имеющимся признакам;
* Каталог должен иметь возможность поэтапного выбора нескольких конкретных характеристик;
* Каталог должен иметь возможность выбора нужной категории квартиры;

После выбора в каталоге подходящей квартиры пользователю предоставляется возможность бронирования.

Требования к бронированию квартиры:

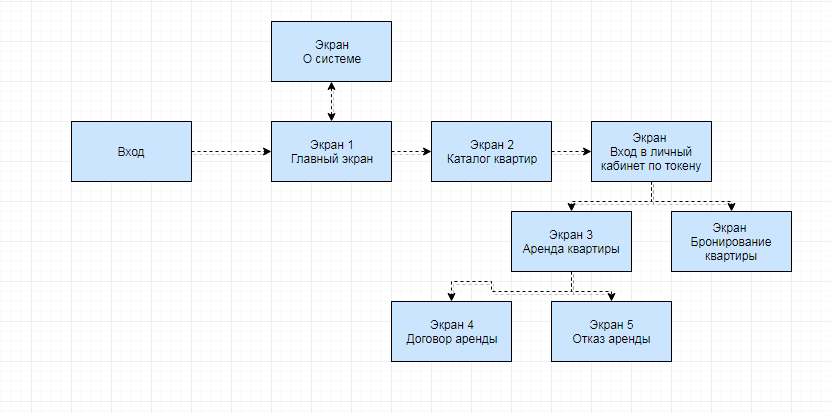
* Бронирование квартиры должно предусматривать все ограничения, т.е. при соблюдении всех ограничений аренды (на полный и неполный дни), у пользователя появляется возможность бронирования.
* Бронирование квартир VIP клиентов также имеет приоритет перед обычными пользователями.
* Бронирование квартир должно устанавливаться на конкретный срок времени.
* Бронирование квартир должно быть лимитировано ( например, не более 2х вариантов у одного пользователя).
* Бронирование квартир должно исключать выбранную квартиру из каталога для других пользователей на срок бронирования.

В системе также предусматривается выбор мелкого ремонта и уборка квартиры.

Требования к заказам мелкого ремонта и уборки квартиры:

* Возможность выбора сторонних компаний по ремонту и уборке.
* Возможность фильтрации сторонних компаний по стоимости, предлагаемым услугам и затраченному времени.
* Возможность бронирования услуг на назначенное время.
* Возможность предоплаты за услугу.
* Возможность выбора/продления услуги на длительный срок.
* Возможность читать/оставлять отзывы о проделанных услугах.

1. Структура системы



Главный экран открывается при каждом запуске приложения. На главном экране можно перейти непосредственно к выбору квартир из каталога либо узнать информацию о самой системе.

Кнопки:

* “О системе” переводит на экран “О системе”.
* ”Выбрать квартиру” переводит на экран ”Каталог квартир”

Экран ”Каталог квартир ” предназначен для выбора характеристик и поиска подходящих вариантов квартиры.

Выпадающие меню:

* “класс комфорта (ремонта)”;
* “год постройки дома”;
* “количество спален”;
* “количество гостиных”;
* “количество сан узлов”; и т.д.

Кнопки:

“Найти” выводит на экран отфильтрованный каталог квартир.

После этого появляются кнопки:

-“Рассчитать аренду”

-”Забронировать жилье”

При нажатии открывается экран ”Вход в личный кабинет” и предлагается предъявить токен, и после этого переход на экраны ”Аренда квартиры” и ”Бронирование квартиры” соответственно.

На экране ”Аренда квартиры” есть:

Выпадающие меню:

* выбор времени
* выбор даты

Кнопка:

Оформить аренду

После нажатия кнопки, появляется экран Договор аренды, в случае успешного соблюдения всех требований, либо экран ”Отказ аренды” в обратном случае.

На экране ”Бронирование квартиры” есть:

Выпадающие меню:

* выбор времени
* выбор даты

Кнопка:

Забронировать аренду

1. Порядок контроля и приёмки системы
   1. Система подвергается испытаниям следующих видов:

* Предварительные испытания
* Опытная эксплуатация
* Приемочные испытания