

Список литературы

1. Xu, B. Empirical Evaluation of rectified activations in convolutional network / B. Xu, N. Wang, T. Chen, M. Li [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1505.00853>. – Дата доступа: 18.10.2021
2. Klambauer, G. Self-Normalizing neural networks / G. Klambauer, T. Unterthiner, A. May, S. Hochreiter [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1706.02515v5>. – Дата доступа: 18.10.2021
3. Ramachandran, P. Swish: a Self-Gated activation function / P. Ramachandran, B. Zoph, Q.V. Le [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1710.05941v2> – Дата доступа: 18.10.2021
4. Person reID baseline pytorch [Электронный ресурс]. – Режим доступа: https://github.com/layumi/Person_reID_baseline_pytorch – Дата доступа: 18.10.2021.

УДК 656.135

РЕАЛИЗАЦИЯ РЕГИСТРАЦИИ ЗАЯВОК ПАССАЖИРОВ ТРАНСПОРТНОЙ СИСТЕМЫ

А. А. Левчук и И. С. Луковец

Брестский государственный технический университет, Брест

Научный руководитель: Шуть Василий Николаевич

доцент БрГТУ, кандидат технических наук

В настоящее время маршрутный транспорт получил широкое распространение среди граждан Республики Беларусь. В связи с этим, увеличилось количество машин на маршрутах; выезжая на маршрут водитель не понимает, будут ли пассажиры на пути его следования.

Нами разрабатывается клиентская и серверная составляющая системы перевозки пассажиров по запросу. В приложении для определения местоположения пользователя планируется использовать qr-код. QR-код, расположенный на каждой остановке общественного транспорта позволяет определить местоположение пассажира и название остановки. Для получения данной информации необходимо считать и расшифровать QR-код. Для обработки QR-кода удобно использовать готовые библиотеки. Для отбора таких библиотек был выполнен их анализ.

Для анализа использованы общедоступные библиотеки Mobile Vision API и Zxing.

В результате анализа была отобрана многоформатная библиотека Zxing для обработки изображений 1D/2D штрих-кодов с открытым исходным кодом [1]. Библиотека реализована на языке Java и может быть использована для различных платформ.

Обработка QR-кода с использованием Zxing в нашем приложении включает следующие действия :

1) Вызов приложения для сканирования qr- кода.

2) Получение входных данных:

а) наведение камеры на qr-код

б) вырезание qr-кода из общего изображения

3) Обработка входных данных:

Шаг 1. Чтение 5 бит системной информации.

Шаг 2. Маска для системной информации.

Шаг 3. Чтение заголовка данных.

Шаг 4. Применение маски к заголовку.

Шаг 5. Чтение данных

4) Передача результатов приложению.

Результатом действий является получение приложением данных для последующей обработки.

Выбрав остановки, данные пользователя отправляются на сервер.

Было принято решение использовать объектно-ориентированную модель для создания нереляционной базы данных. Причинами такого выбора являются требования гибкости и производительности. Указанным требованиям отвечают системы NoSQL баз данных. Помимо этого NoSQL базы данных отказоустойчивы и просты при разработке и масштабировании [4].

В качестве системы управления базой данных определена СУБД MongoDB. Она использует документную модель данных, что обеспечивает, высокую гибкость при разработке и производительность выполняемых запросов. В документной модели данных все записи представлены в виде объектов или словарей, хранящих пары ключ-значение. Схожие объекты в свою очередь объединяются в коллекции, а коллекции – в базу данных. К остальным преимуществам MongoDB относятся: доступ к данным без использования ORM (Object Relational Mappers), дружелюбный к изменениям дизайн, мощная система запросов, простота горизонтального масштабирования. MongoDB уже используется в транспортных системах таких, как Uber или Lyft.

Ниже приведён фрагмент документа для хранения информации на сервере, сгенерированной по запросу пользователя. Документ включает сведения об идентификаторе пользователя, начальной и конечной остановках в выбранном маршруте пользователя, идентификатора(-ов) подходящего маршрута и флага ожидания транспортного средства.

Фрагмент 1. Запрос пользователя на перевозку по маршруту

```
{
  "_id": {
    "$oid": "607437a1f9951287c5440a02"
  },
  "start": "Шоссейная",
  "stop": "Гоголя",
  "routeID": ["2 ВамРад - Бернады"],
  "waitAuto": false
}
```

Требуемые коллекции базы данных реализованы с использованием MongoDB Compass и встроенной оболочки JavaScript. Примерам другой реализованной коллекции является:

коллекция имеющихся пассажирских транспортных средств
коллекция существующих маршрутов

Документ коллекции имеющихся транспортных средств содержит информацию об идентификаторе транспортного средства, маршруте движения транспорта, массиве остановочных пунктов и пассажиров, требующих остановки, количестве пассажиров в транспортном средстве в данный момент и флаге активности транспортного средства.

Фрагмент 2. Документ коллекции имеющихся транспортных средств

```
{
  "_id" : "0001 ВВ-1",
  "routeID" : "2 ВамРад - Автолюбителей",
  "route" : [{
    "name" : "ВамРад",
    "needStop" : 0
  }, {
    "name" : "Новые Задворцы",
    "needStop" : 1
  }],
  "quanPassengers" : 4,
  "workAuto" : false
}
```

Зарегистрировав пользователя на маршруте, сервер отправляет номера маршрутов на которых пользователь может добраться до точки назначения, таким образом можно облегчить поиск маршрута пассажирам и показать нагрузку на маршрут водителям.

Список литературы

1. ZXing ("Zebra Crossing") библиотека сканирования штрих-кодов для Java, Android. [Электронный ресурс]. Режим доступа: <https://github.com/zxing/zxing/> Дата доступа: 08.04.2021
2. Relational vs. NoSQL data. [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/en-us/dotnet/architecture/cloud-native/relational-vs-nosql-data>. – Дата доступа: 08.04.2021.
3. Руководство по MongoDB. Введение. [Электронный ресурс]. – Режим доступа: <https://proselite.net/tutorials/mongodb/introduction/>. – Дата доступа: 14.04.2021.
4. Relational vs. NoSQL data. [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/en-us/dotnet/architecture/cloud-native/relational-vs-nosql-data>. – Дата доступа: 08.04.2021.

УДК 623.462.22

МОДЕЛИРОВАНИЕ РЕГУЛЯТОРА ДЛЯ БОРТОВОЙ СИСТЕМЫ СТАБИЛИЗАЦИИ БЕСПИЛОТНОГО ЛЕТАТЕЛЬНОГО АППАРАТА

А. В. Лопухов, Р. С. Онищук, А. И. Федоров

*Учреждение образования «Военная академия Республики Беларусь», г. Минск
Научный руководитель: А. И. Федоров, кандидат технических наук, доцент*

Проблема синтеза регулятора для беспилотного летательного аппарата (БЛА), обеспечивающая выполнение требований к точности регулирования или попадания установившихся значений регулируемых в математической модели переменных в заданные допуски, при действии на систему внешних и внутренних возмущений, является одной из центральных задач (проблем) в теории и практике автоматического управления.

В процессе управления необходимо задать желаемые переходные процессы замкнутой системы. Такая динамика обусловлена необходимостью разработки наиболее эффективных методов синтеза регуляторов, обеспечивающих желаемые переходные процессы.