

Рисунок 4.2 – Решение дискриминантного анализа в MathCAD (продолжение)

На основе изученных свойств и методов компонента TADOTable среды программирования Delphi можно создать приложение проведения дискриминантного анализа, как показано на рисунке 5. По итогам проведения данного анализа можно обоснованно принимать решение курсанту и преподавателю автошколы в процессе обучения.

Таким образом, синтез автоматизированного проектирования баз данных и математического метода дискриминантного анализа позволит существенно повысить эффективность обучения курсанта, производительность труда сотрудников ООО «Автомобилист» по Оренбургской области и снизить как временные, так и материальные затраты.

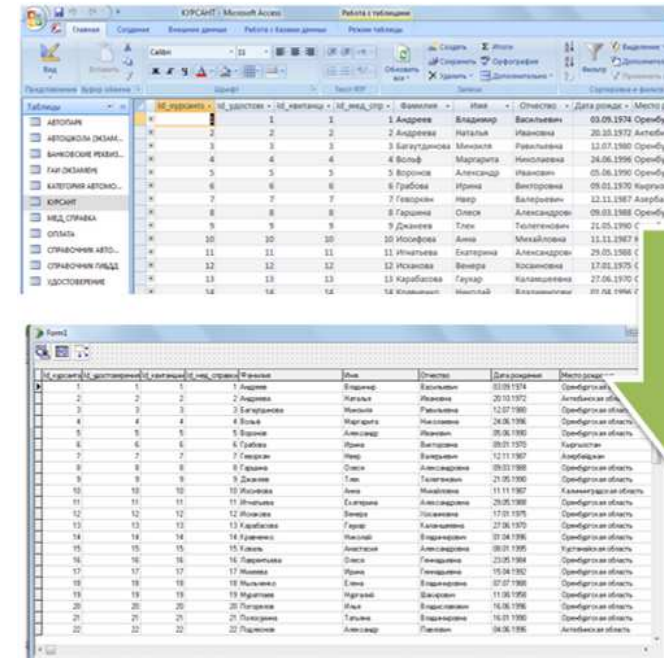


Рисунок 5 – Результат технологии соединения

### Литература

1. Волкова, Т.В. Проектирование и создание БД: учебное пособие / Т.В. Волкова. – Оренбург: ГОУ ОГУ, 2006. – 140 с.
2. Анфилатов, В.С. Системный анализ в управлении: учебное пособие / В.С. Анфилатов, А.А. Емельянов, А. А. Кукушкин. – М.: Финансы и статистика, 2003. – 368 с.

А.Е. Лисогор

### ПРИЛОЖЕНИЕ "ПРИБЫВАЛКА-63" ДЛЯ ПЛАТФОРМЫ J2ME

(Самарский государственный университет путей сообщения)

Практически нет людей, которые не пользуются общественным транспортом. Облегчить перемещения по городу призван Транспортный Оператор Самары – сервис, предоставляющий оперативную информацию о том, какое средство передвижения и когда прибудет на выбранную остановку, информацию о маршрутах, курсирующих по городу и области, а также иные полезные сведения по транспортным вопросам [1][2].



Некоторое время назад в дополнение к уже имеющимся мобильным приложениям для платформ Android, IOS, Blackberry, Windows Phone[1] и веб-версиям сайта Транспортного Оператора Самары [tosamara.ru](http://tosamara.ru) была разработана и выпущена Прибывалка-63 для операционных систем на базе J2ME и Symbian, которая успешно заняла свое место среди остальных программ.

В долгосрочной перспективе сокращение доли J2ME и Symbian неизбежно приведет к уменьшению популярности приложения, но, как показывает практика, пока количество запросов на получение информации о прибытии транспорта держится на определенном уровне и в среднем составляет от 1 до 2 тысяч запросов в день.

В связи с этим развитие приложения не останавливается, его функциональность пополняется новыми возможностями, последней из которых на текущий момент является функция построения маршрута между заданными пользователем точками.

Подобная задача достаточно часто стоит перед жителями крупных городов.

Каждый картографический сервис, будь то Яндекс.Карты[3] или Google Maps[4], предлагает функцию построения маршрутов между точками, указанными пользователем. Однако данные, предоставляемые ими, носят лишь рекомендательный характер, не отражая реальное положение транспорта и его движение по заданному маршруту.

Транспортный Оператор Самары позволяет как построить маршрут, так и получить данные, которые помогут точно спрогнозировать движение пользователя от начальной точки до места назначения. Используя эти сведения, модуль построения маршрута Прибывалки-63 для платформы J2ME сможет определить все интересующие конечного пользователя параметры маршрута, например, время в пути, количество пересадок, расходы на транспорт и так далее, и сообщить пользователю точное время движения[5].

Модуль работает по следующему алгоритму. В связи с тем, что прямое получение координат на платформы J2ME затруднено, приложение отправляет запрос к API геокодирования поисковой системы Спутник[6] на преобразование адресов начальной и конечной точки маршрута.

После получения координат модуль выполняет запрос, включающий в себя также предпочитаемые виды транспорта и критерий оптимальности (например, время, длину или стоимость)[5], к API Транспортного Оператора.

Ответом на запрос является xml-файл, содержащий в себе сведения о длине построенного маршрута, стоимости перемещения, времени, потраченному на его прохождение, а также подробного описания самих перемещений между точками.

Модуль анализирует части маршрута следующим образом:

- Для находящихся в секции «walk» сохраняются значения предыдущей (stopFrom) и следующей остановки (stopTo), время перемещения между ними (time) и комментарий (comment);



- Для находящихся в секции «pass» сохраняются значения начальная (stopFrom) и конечная (stopTo) остановки, все подходящие для перемещения между ними маршруты (routes) транспорта, время перемещения (time) и комментарий (comment).

После этого модуль выполняет запрос прогноза движения[5] для каждого «транспортного» блока, используя значение stopFrom, а затем производит фильтрацию ответа, сохраняя только те маршруты, которые есть в routes и подходящие по времени. Кроме того, запоминается идентификатор первого подходящего транспорта (hullno).

Если hullno существует, модуль выполняет еще один запрос к API Транспортного оператора[5], получая количество остановок, которое нужно проехать на данном транспорте, и время, потраченное на движение. При отсутствии прогноза или подходящего транспорта используются значения по умолчанию, полученные от Транспортного оператора.

После того, как модуль выполнит все рассмотренные операции, происходит итоговая генерация текста, который будет показан пользователю, по следующим шаблонам:

- Пешком ~ 5 мин.  
до ост. Проспект Карла Маркса

- На транспорте:  
Вам подходят:  
Трамвай #7 - через 7 мин.  
Автобус #55 - через 10 мин.

-----  
Проехать 3 остановок ~ 3 мин.  
до ост. Московское шоссе

Кроме того, рассчитывается и отображается на экране итоговое время, представляющее собой сумму всех временных интервалов, затраченных на перемещение по каждому из участков построенного маршрута.

Таким образом, пользователь получает всю интересующую его информацию с максимально уточненным временем движения от начальной до конечной точек маршрута.

### Литература

1. Транспортный Оператор Самары [Электронный ресурс]. URL: <http://tosamara.ru> (дата обращения: 07.03.2016).
2. Сайт Транспортного Оператора Самары [Электронный ресурс]. URL: <https://mymaster.livejournal.com/273735.html> (дата обращения: 05.03.2015).
3. Яндекс.Карты — Википедия [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/Яндекс.Карты> (дата обращения: 13.03.2016).
4. Google Maps — Википедия [Электронный ресурс]. URL: [https://ru.wikipedia.org/wiki/Google\\_Maps](https://ru.wikipedia.org/wiki/Google_Maps) (дата обращения: 13.03.2016).



5. Транспортный Оператор Самары — Программные интерфейсы (API) [Электронный ресурс]. URL: <http://tosamara.ru/api/> (дата обращения: 07.03.2016).
6. Спутник / Карты — Сервис геокодирования [Электронный ресурс]. URL: <http://api.sputnik.ru/maps/geocoder/> (дата обращения: 09.03.2016).

Т.И. Михеева, А.Н. Имамудинов, А.В. Золотовицкий

## МНОГОУРОВНЕВАЯ АРХИТЕКТУРА ВЕБ-ОРИЕНТИРОВАННОЙ ГЕОИНФОРМАЦИОННОЙ СИСТЕМЫ ITSGIS

(Самарский национальный исследовательский университет  
имени академика С.П. Королёва)

В настоящее время, при принятии решения об оптимальном управлении объектами транспортной инфраструктуры (ТрИ), стоит вопрос обеспечения доступа к информации о состоянии объектов ТрИ, которые являются важной и неотъемлемой составляющей транспортного комплекса любого мегаполиса, и обеспечивают его функционирование.

Основной проблемой, возникающей при управлении объектами ТрИ, являются устаревшие методики накопления и обработки информации. Анализ состояния ТрИ осуществляется, в основном, опираясь на бумажные носители: таблицы, ведомости о наличии технических средств организации дорожного движения (ТСОДД), карты и отчеты о полевых работах. В таких условиях лицо, принимающее решение об управлении ТрИ, опирается на устаревшую информацию. Определение взаимного расположения, мониторинг состояния, доступ к атрибутивной информации объектов ТрИ, при такой организации работы, представляется сложным и трудоемким процессом, т.к. плотность и объемы информации достаточно высоки для их ручной обработки. Появляется необходимость в разработке новых методов и средств для учета, оперативного доступа к атрибутивным и пространственным данным, оценки состояния объектов ТрИ.

Совокупность достижений в области интеллектуальных транспортных систем (ИТС), геоинформационных систем (ГИС) и Интернет-технологий, предоставляющих инструменты для удаленного доступа к информационным ресурсам, позволили по-новому решать задачи визуализации, предоставления оперативного доступа к информации о состоянии и дислокации объектов ТрИ.

Разработанная система дислокации объектов ТрИ позволит сформировать представление об актуальном взаимном расположении и состоянии объектов и процессов ТрИ для дальнейшего принятия решения об управлении.

Синтез веб-ориентированных ГИС дислокации и поддержки управления объектов ТрИ на базе ИТС сопряжен с рядом проблем, связанных с построением наиболее эффективной архитектуры, обеспечивающей надлежащий уровень производительности, масштабируемости и надежности системы. В широком смысле построение архитектуры сводится к выбору основных составляющих системы: базовой ГИС-технологии, средства хранения пространственных дан-



ных. В узком смысле – это применение наиболее эффективных архитектурных решений на каждом уровне проектируемой ГИС, где центральное место занимают средства веб-публикации.

Основой разработанной веб-ориентированной системы является многоуровневая архитектура с распределенной на две части (клиентская и серверная) бизнес-логикой. Оставаясь в рамках архитектуры системы, каждая часть имеет более сложную организацию и делится на несколько уровней. На рисунке 1 представлена архитектура веб-ориентированной геоинформационной системы дислокации и поддержки управления объектов транспортной инфраструктуры.

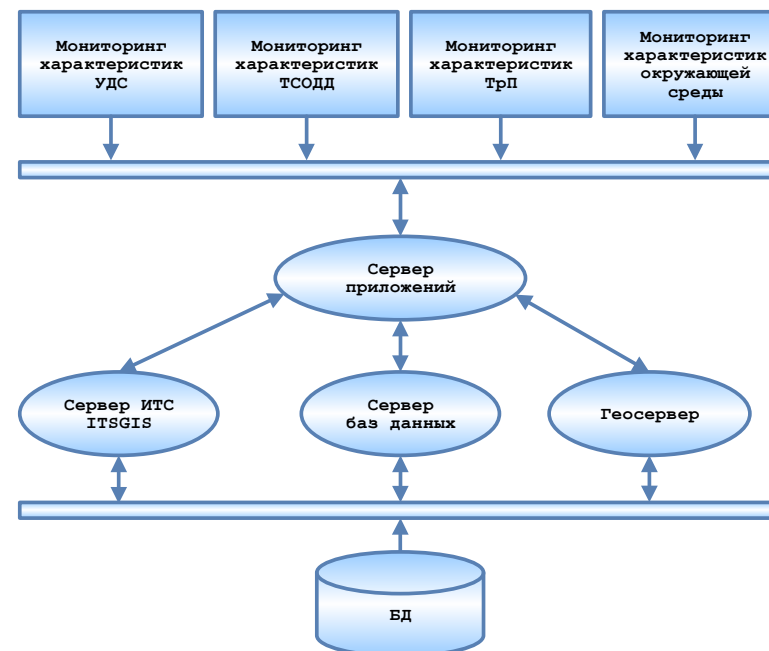


Рисунок 1 – Архитектура системы

Такой подход к организации структуры веб-ориентированной ГИС дислокации объектов ТрИ создает условия для обеспечения централизованной обработки, хранения и доставки пространственных данных через сеть Интернет для удаленных пользователей, решающих задачи справочно-информационного и аналитического обслуживания. Данное архитектурное решение обладает рядом преимуществ:

- выполнение независимо от операционной системы;
- возможность использования на мобильных устройствах;
- максимально быстрое распространение среди клиентов;
- минимальная аппаратная платформа;
- автоматическое обновление версий.