

Федеральное государственное автономное образовательное
учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
институт
Информационные системы
кафедра

УТВЕРЖДАЮ
Зав. кафедрой ИС
_____ С.А. Виденин
«__» _____ 2016 г.

ДИПЛОМНЫЙ ПРОЕКТ

230102.65 Автоматизированные системы обработки информации и управления

«Разработка автоматизированной системы расчета оптимального маршрута для
курьерской компании»

Пояснительная записка

Руководитель

подпись, дата

Л.С. Троценко
инициалы, фамилия

Выпускник

подпись, дата

К.Б. Карманов
инициалы, фамилия

Нормоконтролер

подпись, дата

Л.С. Троценко
инициалы, фамилия

Красноярск 2016

РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка автоматизированной системы расчета оптимального маршрута для курьерской компании» содержит 41 страницу текстового документа, 1 приложение, 11 использованных источников, 10 иллюстраций.

МАРШРУТ, АЛГОРИТМ БЕЛЛМАНА-ФОРДА, КУРЬЕР, ДОСТАВКА ТОВАРОВ, ОПТИМИЗАЦИЯ, АЛГОРИТМ ДЕЙКСТРЫ, АЛГОРИТМ ФЛОЙДА-УОРШАЛЛА

Объектом исследования является: курьерская компания «Миг».

Предметом исследования является: реализация маршрутной сети курьерской компании.

Цели и задачи исследования:

Повысить эффективность процесса доставки с помощью автоматизированной системы расчета оптимального маршрута.

Задачи:

- изучить процесс доставки грузов;
- проанализировать процесс логистики;
- разработать мероприятия по совершенствованию процесса доставки;
- разработать приложение для расчета оптимального маршрута.

Актуальность работы обоснована широким применением информационных технологий в разных сферах деятельности, что повышает эффективность оказываемых услуг, и улучшает клиентоориентированность предприятий в сфере курьерских доставок. Применение современных технологий сокращает временные и трудовые затраты и соответственно делает предприятие рентабельным.

Данная работа прошла апробацию и была внедрена, что подтверждается прилагаемым актом.

					ДП-230102.65-1017437ПЗ			
Ли	Изм.	№ докум.	Подп.	Дата	Разработка автоматизированной системы расчета оптимального маршрута для курьерской компании.	Лит	Лист	Листов
Разраб.		Карманов К.Б.		14.06			2	43
Пров.		Троценко Л.С.		15.06				
Н. контр.		Троценко Л.С.		20.06				
Утв.		Виденин С.А.		23.06				
						Кафедра Информационные системы		

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Характеристика курьерской компании	6
1.1 Структурная схема курьерской компании	6
1.2 Деятельность курьерской компании	7
2 Описание услуг	9
2.1 Функциональное назначение услуг	9
2.2 Виды услуг	10
2.3 Виды доставок	10
3 Анализ алгоритмов для создания автоматизированной системы	12
3.1 Алгоритм Дейкстры	12
3.2 Алгоритм Беллмана-Форда	17
3.3 Выбор алгоритма	22
4 Программная реализация	23
4.1 Инструментарий	23
4.2 Диаграмма сущностей	29
4.3 Выходные данные	30
4.4 Описание приложения	34
4.5 Визуализация результатов работы	39
ЗАКЛЮЧЕНИЕ	40
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	41
ПРИЛОЖЕНИЕ А	42

ВВЕДЕНИЕ

Вполне очевиден тот факт что транспорт имеет огромное влияние на полноценное социально-экономическое развитие страны в целом и регионов в частности. Транспортные услуги предназначены для постоянного и стабильного товарооборота, осуществления целей стратегического характера и перевозок пассажиров.

Решением вышеперечисленных задач занимается транспортная логистика, которая заинтересована в увеличении своей выручки за счет оптимизации транспортной инфраструктуры. Нахождение оптимальных маршрутов для перевозок и выбор оптимального места расположения логистических объектов - основополагающие задачи которые приходится решать транспортной логистике. Решение этих задач приводит к разработке различных приложений для оптимизации бизнес процессов.

В современном мире зачастую приходится пользоваться услугами по доставке различных грузов и корреспонденции. Грузоперевозки выполняют незаменимую работу и давно стали обыденным фактором в жизни современного общества.

На сегодняшний день службы курьерской доставки осуществляют, перевозку и доставку самых разных грузов - начиная с доставки различной корреспонденции до перевозки разно габаритных предметов непосредственно адресату. Стоит заметить что курьерская доставка не ограничивается границами того или иного населенного пункта, а выходит за его границы для обслуживания клиентов из других мест. Основополагающий принцип работы для курьерской службы - это доставка до двери клиента.

Развитие интернет магазинов спровоцировало рост востребованности курьерских услуг у населения. Развитие электронных торговых площадок с каждым днем набирает популярность - что дает значительный импульс для развития курьерских служб.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		4

Любая транспортная компания, которая закрепились на рынке логистических услуг и вышла на определённые финансовые обороты, закономерно начинает задумывается об автоматизации своей работы. У решения автоматизации процесса доставки существуют явные плюсы:

- внедрение новых алгоритмов для оптимизации затрат;
- уменьшение каждодневной нагрузки для сотрудников вследствие чего последует ускорение выполнения работы;
- создание планов работы, их контроль и мониторинг исполнения.

Основные задачи встающие перед автоматизацией - это построение оптимальных маршрутов для перевозок и оценка затраченных на доставку груза временных ресурсов. Решение этих острых задач поднимается в данном проекте, что, несомненно, говорит о его актуальности.

Целью проекта является модернизация процесса доставки путем интегрирования в бизнес процесс автоматизированной системы расчета оптимального маршрута.

Для достижения поставленной цели необходимо решить следующие задачи.

- изучить процесс доставки грузов;
- проанализировать процесс логистики;
- разработать мероприятия по совершенствованию процесса доставки;
- разработать приложение для расчета оптимального маршрута.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		5

1 Характеристика курьерской компании

В процессе выполнения дипломного проекта был проведен мониторинг и проанализирована профессиональная деятельность курьерской компании «Миг».

1.1 Структурная схема курьерской компании

В курьерских компаниях используется линейная система управления. Во главе линейной системы управления генеральный директор, который координирует работу различных подразделений и осуществляет контроль за их работой (Рисунок 1).

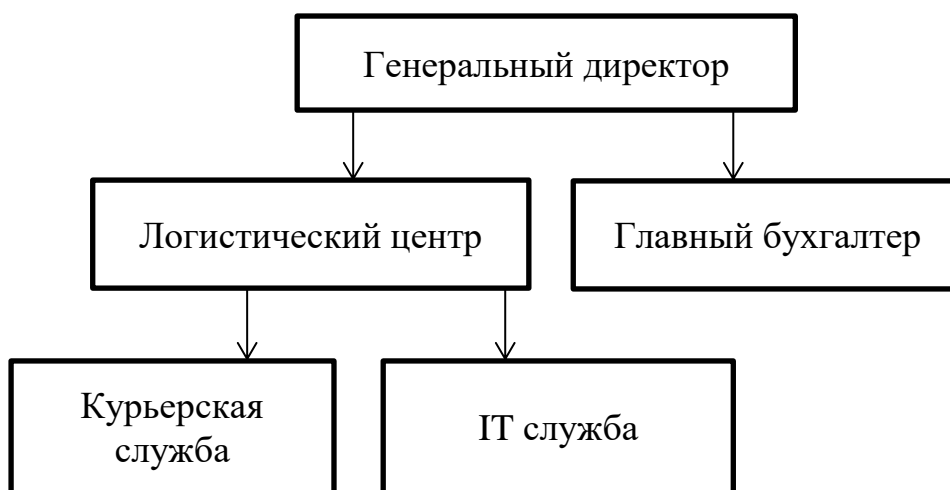


Рисунок 1 - Система управления

Директор курьерской компании несет ответственность за соблюдение всех необходимых требований в работе компании. За наличие необходимых лицензий. Он непосредственно занимается заключением договоров и следит за надлежащим их исполнением. Несет ответственность за качество и размещение рекламы, управляет текущей деятельностью компании. Непосредственно директору подчиняется весь персонал компании.

Генеральному директору подчиняется логистический центр и главный бухгалтер.

Главный бухгалтер осуществляет бухгалтерский и налоговый учет. Несет ответственность за финансовые потоки предприятия, сдачу отчетности, уплату налогов, пенсионных взносов и взносов социального страхования.

Логистический центр, в свою очередь, обеспечивает получение, обработку и хранение информации. На основе полученной информации логистический центр разрабатывает заявку на доставку грузов для курьеров. Несет ответственность за организацию работы курьеров. Контролирует режим работы, принимает заказы, следит за надлежащим исполнением услуг по доставке.

В служебные обязанности входит:

- обеспечение курьеров работой и равномерное распределение нагрузки;
- контроль за работой курьеров;
- осуществляет связь с заказчиками и адресатами;
- мониторинг работы подконтрольных служб.

Курьер непосредственно выполняет заказы, осуществляет доставку/передачу.

IT отдел следит за интеграцией программных продуктов в бизнес процесс курьерской компании. Обеспечивает бесперебойную работу компании в сфере информационной поддержки и своевременно решает возникающие проблемы в сфере IT.

1.2 Деятельность курьерской компании

Главными задачами в деятельности курьерской компании являются:

- приём заказов от заказчика;
- своевременная и надежная доставка заказа адресату;
- разработка спецпредложений для клиентов;
- контроль за качеством услуг;

Изм.	Лист	№ документа	Подпись	Дата

ДП-230102.65-1017437ПЗ

Лист

7

- обучение персонала;
- повышение квалификации персонала;
- расширение дополнительных услуг;
- введение новой автоматизированной системы управления.

Данные мероприятия необходимы для того, чтобы усилить позиции компании на рынке курьерских услуг.

У курьерской компании есть возможности для развития, так как статус компании даёт ей преимущество:

- рост количества клиентов доверяющих компании;
- привлечение требовательных клиентов из числа государственных компаний.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		8

2 Описание услуг

Курьерская компания - это предприятие, специализирующееся исключительно на услугах доставки грузов различными объемами и разными видами транспорта. Технология и реализация профессиональной деятельности курьерской компании распространяется в транспортной логистике. У транспортной организации существует целая сеть партнеров, пользующихся ее услугами, как аутсорсинговыми. Курьеры такой компании обычно делятся на людей, работающих с разными видами доставки - на дальние и близкие расстояния, с различными грузами - от бумажных до вещественных. Обычно и цена такой доставки зависит от места, где находится адресат. Но и то, что эти услуги платные, не делает их менее популярными, ведь они значительно экономят время.

2.1 Функциональное назначение услуг

На сегодняшний день курьерская компания предлагает для своих клиентов целый спектр услуг по обычной и срочной доставке различной корреспонденции и грузов всех видов. Зона покрытия включает в себя город Усть-Илимск, а также населенные пункты Усть-Илимского района. Курьерская компания обеспечивает доставку корреспонденции по принципу «из рук в руки». Все передвижения курьеров и отправления строго контролируются от самого момента приема груза и до доставки адресату. Помимо стандартного комплекса курьерских услуг, так же присутствует и упаковка, подтверждение о доставке, страхование. Доставка корреспонденции и грузов осуществляется автомобильным транспортом различной грузоподъемности и курьерами.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		9

2.2 Виды услуг

Курьерская служба в настоящее время оказывает следующие услуги: доставку по городу Усть-Илимску и Усть-Илимскому району:

- деловой корреспонденции;
- документации;
- компьютерной техники и расходных материалов;
- счетов;
- каталогов;
- буклетов;
- пригласительных;
- писем и открыток;
- посылок и бандеролей;
- подарков;
- цветов;
- товаров из интернет-магазинов;
- других материалов.

К дополнительным возможностям стоит отнести бесплатный выезд курьера к заказчику в пределах города Усть-Илимска, а так же СМС уведомление о доставленном грузе.

2.3 Виды доставок

а) Стандартная доставка.

Стандартная доставка осуществляется, как правило, на следующий день после приема отправления. После исполнения заказа сотрудник логистического центра уведомляет заказчика об исполнении заказа посредством СМС уведомления либо по электронной почте.

б) Срочная доставка.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		10

В свою очередь срочная доставка осуществляется в течение 2-3 часов только по городу Усть-Илимску и его району. При условии нахождения отправителя и получателя в пределах города доставка произойдет в течение 1 часа. Срок выполнения заказа исчисляется от момента принятия заказа курьерской службой. Курьер прибывает к заказчику в заранее оговоренное заказчиком время, при условии размещения заказа не позднее, чем за 3 часа до времени прибытия курьера. После исполнения заказа сотрудник логистического центра уведомляет заказчика об исполнении заказа посредством СМС уведомления либо по электронной почте.

в) Экономная доставка.

Экономная доставки осуществляется в течении 3 дней После исполнения заказа сотрудник логистического центра уведомляет заказчика об исполнении заказа посредством СМС уведомления либо по электронной почте.

г) Доставка больших объемов корреспонденции.

Рассылка приглашений, поздравлений, счетов, и т.д. партнерам по бизнесу, друзьям.

д) Доставка цветов.

е) Работа с сервисным центром по ремонту ПК.

Доставка в сервисный центр картриджей для заправки, компьютерной периферии для ремонта. Возврат вышеперечисленных вещей заказчику после ремонта.

ж) Дополнительные услуги.

Аренда водителя с автомобилем либо пешеходного курьера.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		11

3 Анализ алгоритмов для создания автоматизированной системы

Основной задачей транспортной компании является доставка груза в минимально возможное время с наименьшими затратами ресурсов. Следовательно особо остро стоит задача выбора оптимального маршрута.

В зависимости от маршрутной сети применяются различные алгоритмы:

- Дейкстры;
- Беллмана-Форда;
- ветвей и границ;
- Фолда-Уоршела.

Часть алгоритмов предназначены для расчета маршрута по замкнутому контуру, другие для расчета пути между двумя заданными точками.

Ключевую роль в создании системы занимает выбор алгоритма, по которому впоследствии и будет работать приложение. Это является первоочередной задачей. Что бы взвесить все плюсы и минусы того или иного алгоритма, необходимо их детально разобрать и провести сравнительный анализ.

3.1 Алгоритм Дейкстры

Рассмотрим алгоритм Дейкстры.

Для примера был взят ориентированный граф G:

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		12

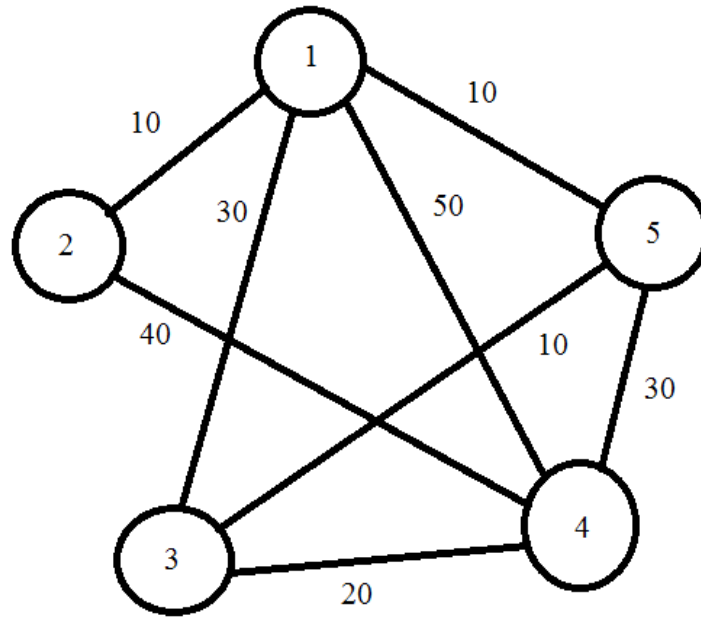


Рисунок 2- Граф G

В свою очередь тот же самый граф возможно представить в виде матрицы С (Рисунок 3).

	1	2	3	4	5
1		10	30	50	10
2					
3					10
4		40	20		
5	10		10	30	

Рисунок 3 - Матрица С

Вершина 1 была выбрана в качестве источника. Это означает что искать кратчайшие маршруты необходимо из вершины 1 в вершины 2, 3, 4 и 5 соответственно. Данный алгоритм поочередно перебирает все вершины графа и попутно назначает им метки, которые являются известным минимальным расстоянием от вершины источника до конкретной вершины. Подробнее

данный алгоритм рассмотрен ниже на примере. 1-й вершине присваивается метка равная 0, потому как эта вершина является источником. Остальным же вершинам присваиваются метки равные бесконечности.

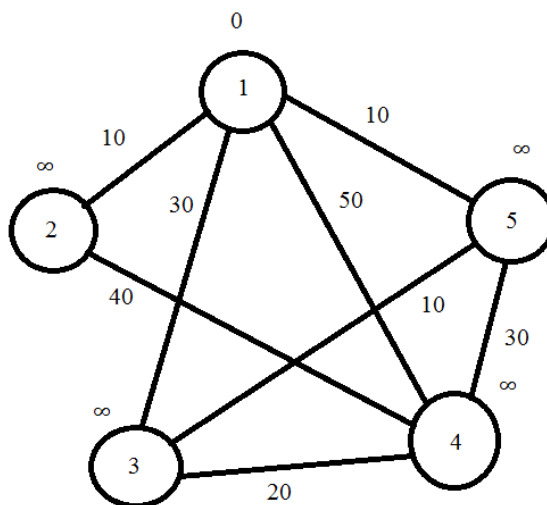


Рисунок 4 - Присваивание вершине 1 метки 0

Далее необходимо выбрать такую вершину W , которая имеет минимальную метку (сейчас это вершина 1) и рассмотреть все вершины в которые из вершины W есть путь, не содержащий вершин посредников. Каждой из рассмотренных вершин назначается метка равная сумме метки W и длины пути из W в рассматриваемую вершину, но только в том случае, если полученная сумма будет меньше предыдущего значения метки. Если же сумма не будет меньше, то предыдущая метка остается без изменений.

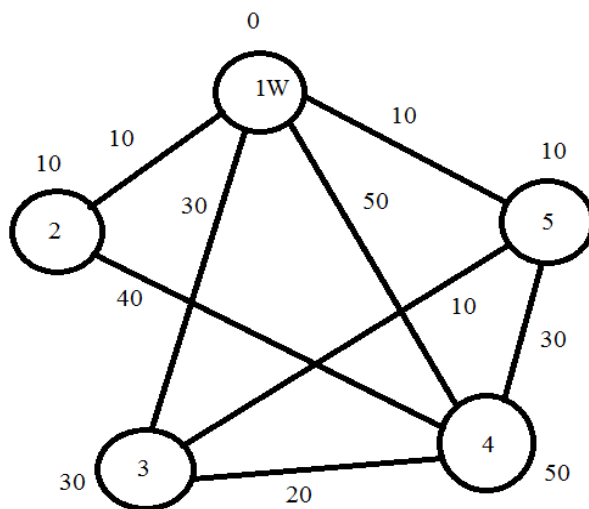


Рисунок 5 - Назначение меток вершинам

После того как были рассмотрены все вершины, в которые есть прямой путь из W , вершина W отмечается как посещенная, а далее выбирается из ещё не посещенных такая, которая имеет минимальное значение метки, она и будет следующей вершиной W . В данном случае это вершина 2 или 5. Если есть несколько вершин с одинаковыми метками, то не имеет значения какую из них будет выбрана как W . В данном случае это вершина 2. Но из нее нет ни одного исходящего пути, поэтому необходимо отметить эту вершину как посещенную и перейти к следующей вершине с минимальной меткой. На этот раз только вершина 5 имеет минимальную метку. На следующем этапе необходимо рассмотреть все вершины в которые есть прямые пути из 5, но которые ещё не помечены как посещенные. Следует найти сумму метки вершины W и веса ребра из W в текущую вершину, и если эта сумма будет меньше предыдущей метки, то заменить значение метки на полученную сумму.

Изм.	Лист	№ документа	Подпись	Дата

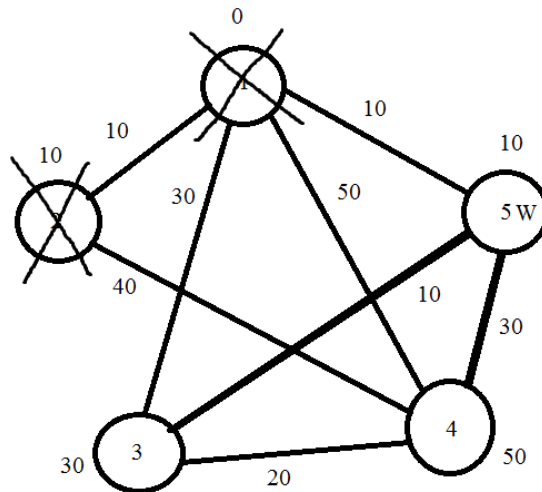


Рисунок 6 - Нахождение более коротких маршрутов в вершины 3 и 4

Исходя из рисунка можно увидеть, что метки 3-ей и 4-ой вершин стали меньше, то есть был найден более короткий маршрут в эти вершины из вершины источника. Далее отмечается 5-ю вершина как посещенная и выбирается следующая вершина, которая имеет минимальную метку. Затем все перечисленные выше действия повторяются до тех пор, пока есть не посещённые вершины. Выполнив все действия получится такой результат:

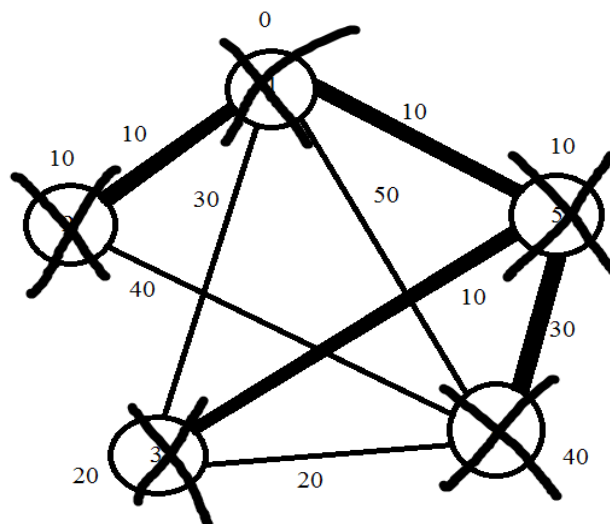


Рисунок 7 - Результат выполнения алгоритма

Изм.	Лист	№ документа	Подпись	Дата

Кратчайшее расстояние от начала графа (1) до каждого узла можно представить в виде таблицы:

Таблица 1 – Результат выполнения алгоритма Дейкстры

Узел	Путь	Расстояние
2	1→2	10
3	1→5→3	20
4	1→5→4	40
5	1→4	10

Сложность алгоритма Дейкстры $O(V^2E)$, где V - количество вершин; E - количество ребер.

3.2 Алгоритм Беллмана-Форда

Алгоритм Беллмана-Форда - это алгоритм предназначенный для поиска кратчайшего пути во взвешенном графе. За время $O(|V| \times |E|)$ алгоритм находит кратчайшие пути от одной вершины графа до всех остальных. В отличие от того же алгоритма Дейкстры, алгоритм Беллмана-Форда может работать с рёбрами с отрицательным весом.

Алгоритм носит имя двух американских учёных: Ричарда Беллмана (Richard Bellman) и Лестера Форда (Lester Ford). Форд фактически изобрёл этот алгоритм в 1956 г. при изучении другой математической задачи, подзадача которой свелась к поиску кратчайшего пути в графе, и Форд дал набросок решающего эту задачу алгоритма. Беллман в 1958 г. опубликовал статью, посвящённую конкретно задаче нахождения кратчайшего пути.

Формулировка задачи

Для примера был взят ориентированный или неориентированный граф G со взвешенными рёбрами. Длинной пути называется сумма всех ребер,

входящих в этот путь. Требуется найти кратчайшие пути от выделенной вершины s до всех остальных вершин графа.

Необходимо заметить, что кратчайших путей может не существовать. Так, в графе, содержащем цикл с отрицательным суммарным весом, существует сколько угодно короткий путь от одной вершины этого цикла до другой (потому как каждый обход цикла уменьшает длину пути). Если сумма весов рёбер цикла отрицательна, то такой цикл называется отрицательным.

Решение задачи на графе без отрицательных циклов

Поставленная задача будет решаться на графе, в котором заведомо нет отрицательных циклов.

Для нахождения кратчайших путей от одной вершины до всех остальных, необходимо воспользоваться методом динамического программирования. На первом этапе строится матрицу A_{ij} , элементы которой будут обозначать следующее: A_{ij} - это длина кратчайшего пути из s в i , содержащего не более j рёбер.

Путь, содержащий 0 рёбер, существует только до вершины s . Таким образом, A_{i0} равно 0 при $i = s$, и $+\infty$ в противном случае.

Далее рассматриваются все пути из s в i , содержащие ровно j рёбер. Каждый такой путь в графе есть путь из $j - 1$ ребра, к которому добавлено последнее ребро. Если по пути длины $j - 1$ все данные уже подсчитаны, то не составляет особого труда определить j -й столбец матрицы.

Алгоритм поиска длин кратчайших путей в графе без отрицательных циклов выглядит таким образом:

```
for  $v \in V$ 
do  $d[v] \leftarrow +\infty$ 
 $d[s] \leftarrow 0$ 
for  $i \leftarrow 1$  to  $|V|-1$ 
do for  $(u,v) \in E$ 
if  $d[v] > d[u] + w(u,v)$ 
```

then $d[v] \leftarrow d[u] + w(u, v)$

return d

Здесь V - множество вершин графа G , E - множество его рёбер, а w - весовая функция, заданная на ребрах графа (возвращает длину дуги ведущей из вершины v в u), d - массив, содержащий расстояния от вершины s до любой другой вершины.

Поскольку известно что кратчайший путь не может содержать большее число ребер, то Внешний цикл выполняется $|V| - 1$ раз. В противном случае он будет содержать цикл, который абсолютно точно можно удалить.

Вместо массива d можно хранить всю матрицу A , но это требует $O(V^2)$ памяти. Благодаря этому факту можно вычислить не только их длины, но и конечно же сами кратчайшие пути. Для этого заводится матрица P_{ij} .

Если элемент A_{ij} содержит длину кратчайшего пути из s в i , содержащего j рёбер, то P_{ij} содержит предыдущую вершину до i в одном из таких кратчайших путей (ведь их может быть несколько).

После преобразований алгоритм Беллмана-Форда принимает следующий вид:

for $v \in V$

for $i \leftarrow 0$ to $|V| - 1$

do $A_{ij} \leftarrow +\infty$

$A_{s0} \leftarrow 0$

for $i \leftarrow 1$ to $|V| - 1$

do for $(u, v) \in E$

if $A_{vi} > A_{u, i-1} + w(u, v)$

then $A_{vi} \leftarrow A_{u, i-1} + w(u, v)$

$P_{vi} \leftarrow u$

После выполнения этого алгоритма элементы A_{ij} содержат длины кратчайших путей от s до i с количеством ребер j , и из всех таких путей

необходимо выбрать самый короткий. А сам кратчайший путь до вершины i с j ребрами восстанавливается следующим образом:

```
while  $j > 0$ 
```

```
 $p[j] \leftarrow i$ 
```

```
 $i \leftarrow P_{ij}$ 
```

```
 $j \leftarrow j - 1$ 
```

```
return  $p$ 
```

Граф с отрицательными циклами

Алгоритм Беллмана-Форда позволяет довольно просто и быстро определить, присутствует ли в графе G отрицательный цикл, который достигается из вершины s . Для этого достаточно произвести внешнюю итерацию цикла не $|V|-1$, а ровно $|V|$ раз. Если при исполнении последней итерации длина кратчайшего пути до какой-либо вершины строго уменьшилась, то этот факт говорит о том что в графе есть отрицательный цикл, достижимый из s . Основываясь на этом выводе можно предложить следующую оптимизацию: отслеживать изменения в графе и, как только они закончатся, сделать выход из цикла, потому как дальнейшие итерации будут бессмысленны.

Сложность алгоритма Беллмана-Форда - $O(V \cdot E)$, где V - это количество вершин, а E - количество ребер.

3.3 Алгоритм Флойда-Уоршалла

Алгоритм Флойда - Уоршелла - это динамический алгоритм, служащий для нахождения кратчайших расстояний между всеми вершинами взвешенного ориентированного графа. Алгоритм был разработан в 1962 году Робертом Флойдом и Стивеном Уоршеллом.

Существует более строгая формулировка этой задачи и звучит она следующим образом: есть ориентированный граф $G = (V, E)$ каждой дуге $v \rightarrow w$ этого графа сопоставлена неотрицательная стоимость $C[v, w]$. Общая задача

									Лист
									20
Изм.	Лист	№ документа	Подпись	Дата	ДП-230102.65-1017437ПЗ				

нахождения кратчайших путей заключается в нахождении для каждой упорядоченной пары вершин v, w) любого пути от вершины v в вершины w , длина которого минимальна среди всех возможных путей от v к w .

Для определенности можно предположить, что вершины графа последовательно пронумерованы от 1 до n . Алгоритм Флойда использует матрицу A размера $n * n$, в которой вычисляются длины кратчайших путей. Вначале $A[i, j] = C[i, j]$ для всех $i \neq j$. Если дуга $i - j$ отсутствует, то $C[i, j] = \text{infinity}$. Каждый диагональный элемент матрицы A равен 0.

Над матрицей A выполняется n итераций. После k -й итерации $A[i, j]$ содержит значение наименьшей длины путей из вершины i в вершину u , которые не проходят через вершины с номером, большим k . Другими словами, между концевыми вершинами пути i и u могут находиться только вершины, номера которых меньше или равны k . На k -й итерации для вычисления матрицы A применяется следующая формула:

$$A_k[i,j]=\min(A_{k-1}[i,j], A_{k-1}[i,k]+A_{k-1}[k,j]) \quad (1)$$

Нижний индекс k обозначает значение матрицы A после k -й итерации, но это не означает, что существует n различных матриц, этот индекс используется для сокращения записи. Для вычисления $A_k[i, j]$ проводится сравнение величины $A_{k-1}[i, j]$ (т.е. стоимость пути от вершины i к вершине j без участия вершины k или другой вершины с более высоким номером) с величиной $A_{k-1}[i, k] + A_{k-1}[k, j]$ (стоимость пути от вершины i до вершины k плюс стоимость пути от вершины k до вершины j). Если путь через вершину k дешевле, чем $A_{k-1}[i, j]$, то в этом случае величина $A_k[i, j]$ изменяется.

Очевидно, что время выполнения данной программы имеет порядок $O(V^3)$, по той простой причине, что в ней практически нет ничего, кроме вложенных друг в друга трех циклов. В свою очередь доказательство «правильности»

работы этого алгоритма также очевидно и выполняется с помощью математической индукции по k , показывая, что на k -й итерации вершина k включается в путь только в том случае, когда новый путь оказывается короче старого.

3.3 Выбор алгоритма

После проведения детального разбора всех перечисленных алгоритмов, становится возможно провести сравнительный анализ. Алгоритм Дейкстры находит кратчайшее расстояние от одной из вершин графа до всех остальных. Этот алгоритм работает только для графов в которых отсутствуют рёбра отрицательного веса. Данный алгоритм довольно широко применяется в программировании и информационных технологиях.

Алгоритм Беллмана-Форда используется для решения аналогичной задачи, если граф может содержать рёбра с отрицательным весом.

В свою очередь алгоритм Флойда-Уоршалла используется для нахождения кратчайших расстояний между всеми парами вершин.

После проведенного выше анализа алгоритмов, было принято решение использовать алгоритм Беллмана-Форда, так как он имеет наименьшую сложность $O(V * E)$. Это позволит системе быть более гибкой и производительной. Что, в свою очередь, призвано повысит эффективность работы курьерской службы.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		22

4 Программная реализация

После того как был проведен сравнительный анализ алгоритма, был выбран оптимальный алгоритм для автоматизированной системы. На следующем этапе необходимо выбрать язык программирования, на котором будет реализована система. Выбор языка программирования не менее важный этап чем выбор алгоритма. На сегодняшний день существуют различные программные продукты для реализации данной системы.

4.1 Инструментарий

Pascal - это язык программирования высокого уровня. Он был очень широко распространен, но в настоящее время идет ориентация на объектно-ориентированные и другие языки. Такие как: C# и Java имеют большую востребованность и эффективность написания кода в отличие от Паскаля. Его изучают не столько для применения, как для обучения в учебных заведениях. Потому что он прост для понимания и нанести вред системе от некорректного кода пользователя не может.

Совместно с числовой информацией в Паскале используется и алфавитно-цифровая или символьная информация, которая включает в себя заглавные, строчные буквы, цифры от 0 до 9 и вспомогательные символы. Для реализации описания символьных переменных используется тип данных CHAR или STRING.

Первоначальная цель разработки языка обосновывалась необходимостью инструмента для обучения программированию как систематической дисциплине. Pascal принадлежит к Algol подобным языкам программирования, по той причине что использует семантику Algol. Однако Pascal имел существенное усовершенствование - жесткую типизацию. То есть для переменных необходимо заранее определять тип данных и принадлежность к определенному правилу (одновременно указывались

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		23

правила, по которым типы считались одинаковыми). Это усовершенствование значительно улучшило стиль программирования, поскольку существенную часть ошибок удавалось обнаружить еще на этапе компиляции - что, в свою очередь, увеличивает надежность программ.

Однако язык разрабатывался как исследовательский проект и первоначальный Pascal был мало пригоден для написания больших проектов, поскольку программу нельзя было составить из нескольких программных частей - просто не было предусмотрено такой возможности. Но этот язык программирования быстро завоевала популярность в учебных заведениях при изучении программирования. А когда появились диалекты языка где возможно было отдельное компилирование программных частей - Pascal стал средством написания больших программных систем.

C++ компилируемый строго типизированный язык программирования общего назначения. Поддерживает разные парадигмы программирования: процедурную, обобщенную, функциональную; наибольшее внимание уделено поддержке объектно-ориентированного программирования.

Разработка языка началась в 1979 году. Целью создания C++ было дополнение C возможностями, удобными для масштабной разработки ПО, с сохранением гибкости, скорости и портативности C. Вместе с тем создатели C++ стремились сохранить совместимость с C: синтаксис первого основан на синтаксисе последнего, и большинство программ на C будут работать и как C++. Изначально новый язык назывался «C с классами», но затем имя было изменено на C++ это должно было подчеркнуть как его происхождение от C, так и его превосходство над последним.

Первый выпуск C++ для коммерческого использования состоялся в 1985 году, вместе с публикацией книги «The C++ Programming Language», которая на долгое время стала его неофициальным стандартом. В 1989 году вышла вторая версия языка в сопровождении книги «The Annotated C++ Reference Manual».

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		24

В 1990-х годах язык стал одним из наиболее широко используемых языков программирования общего назначения. Первым официальным стандартом языка стал ISO/IEC 14882:1998, более известный как C++98. В 2003 году была принята его дополненная версия, C++03, а в 2005 году был опубликован «Library Technical Report 1» (сокращенно TR1) - документ, описывающий расширения стандартной библиотеки. TR1 не является стандартом, но большинство актуальных компиляторов C++ поддерживает его. Наконец, в 2011 году был принят текущий стандарт, C++11.

Нововведениями C++ в сравнении с C являются:

-поддержка объектно-ориентированного программирования через классы. C++ предоставляет все четыре возможности ООП: абстракцию, инкапсуляцию, наследование (в том числе и множественное) и полиморфизм;

-поддержка обобщённого программирования через шаблоны функций и классов;

-стандартная библиотека C++ состоит из стандартной библиотеки C (с некоторыми модификациями) и библиотеки шаблонов (Standard Template Library, STL), которая предоставляет обширный набор обобщенных контейнеров и алгоритмов;

-дополнительные типы данных;

-обработка исключений;

-виртуальные функции;

-пространства имён;

-встраиваемые (inline) функции;

-перегрузка (overloading) операторов;

-перегрузка имён функций;

-ссылки и операторы управления свободно распределяемой памятью.

JavaScript это полноценный динамический язык программирования, который применяется к HTML документу, и может обеспечить динамическую интерактивность на веб-сайтах. Его разработал Brendan Eich, сооснователь проекта Mozilla, Mozilla Foundation и Mozilla Corporation.

						ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата			25

С помощью JavaScript можно реализовать очень многое. От простых функций, таких как карусели, галереи изображений, изменяющихся макетов, и отклика на нажатие кнопок, до игр, анимированной 2D и 3D графики, полномасштабных приложений с базами данных и многое другое.

-JavaScript дает html-дизайнерам инструмент программирования. HTML-разработчики обычно не являются программистами, но JavaScript - язык с очень простым синтаксисом. Каждый может добавить небольшие участки (снипеты) кода в свою html-страницу.

-JavaScript может вставить динамический текст в html-страницу.

-JavaScript может реагировать на события - JavaScript может быть настроен на выполнение какого-либо сценария, когда будет выполнено какое-либо действие, например, окончание загрузки страницы или действия пользователя.

-JavaScript может находить и изменять html-элементы.

-JavaScript может читать и изменять содержимое html-элементов.

-JavaScript может быть использован для проверки введенных данных, для проверки правильности введенных данных перед их отправкой на сервер. Это предохраняет сервер от излишних нагрузок.

-JavaScript может быть использован для определения браузера пользователя, может загрузить страницу, специально созданную для этого браузера.

-JavaScript может быть использован для создания cookies (куки).

-JavaScript может быть использован для хранения и поиска информации в компьютере пользователя.

-JavaScript сам по себе довольно компактный, но очень гибкий, и разработчиками написано много инструментов поверх основного JavaScript языка, которые разблокируют огромное количество дополнительных функций с очень небольшим усилием.

К ним относятся:

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		26

-Программные Интерфейсы приложения (APIs) встроенные в веб-браузерах, обеспечивающие различные функциональные возможности, такие как, динамическое создание HTML и установку CSS стилей, захват и манипуляция видеопотоком, работа с веб-камерой пользователя или генерация 3D графики и аудио сэмплов.

-Сторонние API позволяют разработчикам внедрять функциональность в свои сайты от других разработчиков, таких как Twitter или Facebook.

-Сторонние фреймворки и библиотеки вы можете применить к вашему HTML, что позволит вам ускорить создание сайтов и приложений.

PHP - это скриптовый язык программирования, применяющийся для создания сайтов. Важное его достоинство языка php - это создания динамических веб-сайтов, работа с базами данных (mysql).

Язык PHP - это скриптовый язык программирования, имеющий:

- открывающиеся и закрывающиеся теги;
- переменные;
- функции, которые могут принимать значения и отдавать полученные значения;
- условия, циклы.

Это практически система управления временными данными, которые могут быть довольно больших размеров и хранятся на сервере, а не на компьютере пользователя. Еще одной особенностью языка является способность поддерживать работу сессий и Cookies, что делает работу приложений более персонализированной и удобной для пользователя. Без сессий в PHP практически невозможно управлять динамическими сайтами и приложениями, особенно, если на сайте существует система регистраций, обновлений параметрами пользователя, генерацией страниц сайта, блоками комментариев и особенно системой администрирования.

База данных - это представленная в объективной форме совокупность разных материалов (к примеру: статей, расчётов, нормативных актов,

									Лист
									27
Изм.	Лист	№ документа	Подпись	Дата	ДП-230102.65-1017437ПЗ				

судебных решений и иных подобных материалов), систематизированных так, чтобы эти материалы могли быть найдены и обработаны с помощью ЭВМ.

База данных в веб программировании - это место хранения данных в таблицах.

Для того, чтобы подключиться к базам данных, необходимо знать название БД, пользователя БД и пароль к БД.

SQL - Structured Query Language (Структурированный Язык Запросов) является общим стандартом доступа к базам данных. Компьютерный язык, основанный на стандарте ANSI, предназначенный для доступа и управления базами данных. Команды SQL используются для извлечения и обновления записей в базах данных.

Базы данных чаще всего содержат одну или несколько таблиц. Каждая ячейка идентифицируется по названию (например, «Friends» (Друзья) или «Orders» (Заказы)). Таблицы содержат записи с данными. SQL работает с такими системами управления БД (СУБД), как MS Access, DB2, Informix, MS SQL Server, Oracle, Sybase и др

SQL предназначен для выполнения запросов. Кроме того в SQL входит синтаксис для обновления, вставки и удаления данных. Этот синтаксис вместе с командами обновления формирует язык управления данными (DML):

- SELECT - извлекает данные из таблицы БД;
- UPDATE - обновляет данные в таблице БД;
- DELETE - уничтожает данные в таблице БД;
- INSERT INTO - вставляет новые данные в таблицу БД.

DDL является частью SQL, которая управляет созданием и удалением таблиц в БД, Кроме того, с помощью DDL мы можем назначать индексы (ключевые слова), налаживать взаимосвязи между таблицами и накладывать ограничения на таблицы БД.

Важнейшими командами DDL являются следующие команды:

- CREATE TABLE - создание новой таблицы;
- ALTER TABLE - изменение существующей таблицы;

									Лист
									28
Изм.	Лист	№ документа	Подпись	Дата					

-DROP TABLE - удаление таблицы;

-CREATE INDEX - создание индекса (ключевого слова для облегчения поиска);

-DROP INDEX - удаление индекса.

Реализация алгоритма Беллмана-Форда доступна на любом языке программирования, в том числе и на Pascal. Но данный язык не поддерживает работу с объектами и визуализацию результатов работы.

Использование объектно-ориентированное программирование C++ позволяет работать с объектами, поддерживает графику, но используется только для создания десктопных приложений и не может применяться для web интерфейсов.

Язык Java Script не поддерживает технологию обработки баз данных, выполняется на стороне клиента и следовательно может быть использоваться для расчета кратчайшего пути, но потребует ручного ввода всех данных.

Исходя из вышперечисленного, выбор программных средств был сделан в пользу языка программирования PHP и языка запросов SQL.

4.2 Диаграмма сущностей

Диаграммы «сущность - связь» (EntityRelationDiagram) предназначены для того, чтобы можно было взглянуть на систему в целом. На такой диаграмме рассматриваются сущности со всеми атрибутами и взаимосвязь между этими сущностями. Существует набор стандартных обозначений и правил для того, чтобы можно было верно представить данные.

ERD-диаграммы могут графически описывать разные подзадачи модели данных и показывать как они связаны между собой. Таким образом, можно наглядно рассмотреть каждую сторону проекта и сделать процесс проектирования более простым.

Для того, чтобы разобраться как строятся эти диаграммы, необходимо понять что подразумевается под сущностью и связью.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		29

Сущность это конечное множество реальных или абстрактных объектов, которые объединены общими характеристиками и обладают одними и теми же атрибутами (свойствами). Каждый объект может относиться только к одной сущности и при этом он должен иметь уникальное имя и что-то, что будет отличать его от других объектов данной сущности.

Связь соединяет между собой сущности, являясь отношением или ассоциацией между этими сущностями. Связи бывают трех видов: «один к одному», «один ко многим», «многие ко многим». Для того чтобы отметить эти виды на диаграмме нужно поставить 1(один) или n(многие) над стрелочками (в начале стрелки и в конце).

Диаграмма сущностей представлена на рисунке 8.

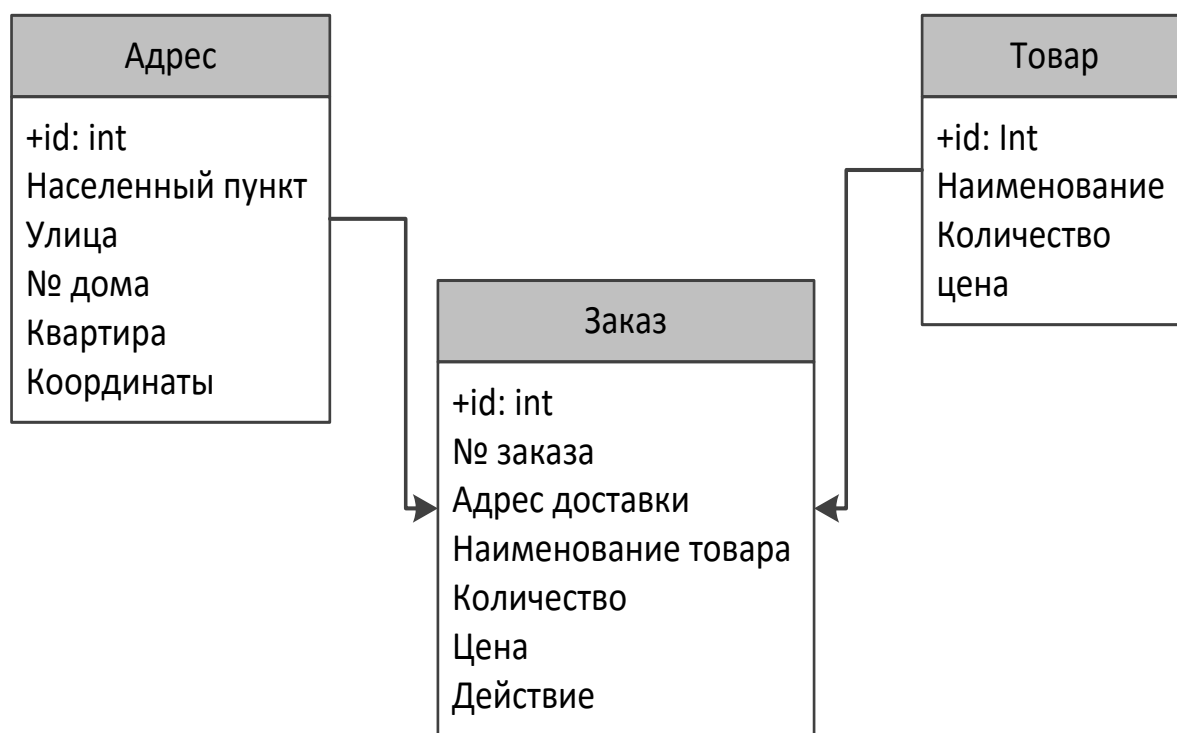


Рисунок 8 - Диаграмма сущностей

4.3 Выходные данные

Для реализации программы необходимо представить карту города Усть-Илимска в виде ориентированного графа. Что, в свою очередь и было

проделано. Карта города была разбита на отдельные сектора, каждый из которых принят за вершину ориентированного графа.

Процесс создания и изменения интуитивно понятен. Визуальное представление является очень понятной формой представления графа, также можно увидеть результат работы алгоритма в визуальной форме. Результат работы можно сохранить в файле изображения. Для большей наглядности в процессе построения графов можно разместить подписи к элементам.

Для редактирования можно использовать различные методы: визуально редактировать или редактировать матрицу смежности графа. Также присутствует возможность создания графа только по матрице смежности.

					ДП-230102.65-1017437ПЗ	Лист
Изм.	Лист	№ документа	Подпись	Дата		31



Рисунок 9 - Карта правобережной части города

Изм.	Лист	№ документа	Подпись	Дата

ДП-230102.65-1017437ПЗ

Лист

32



Рисунок 10 - Карта левобережной части города

Изм.	Лист	№ документа	Подпись	Дата

ДП-230102.65-1017437ПЗ

Лист

33

Координаты объектов с карты помещаются в базу данных (рассмотренную ранее) и в модуль расчета маршрута. Обработка данных осуществляется с помощью разрозненной функции.

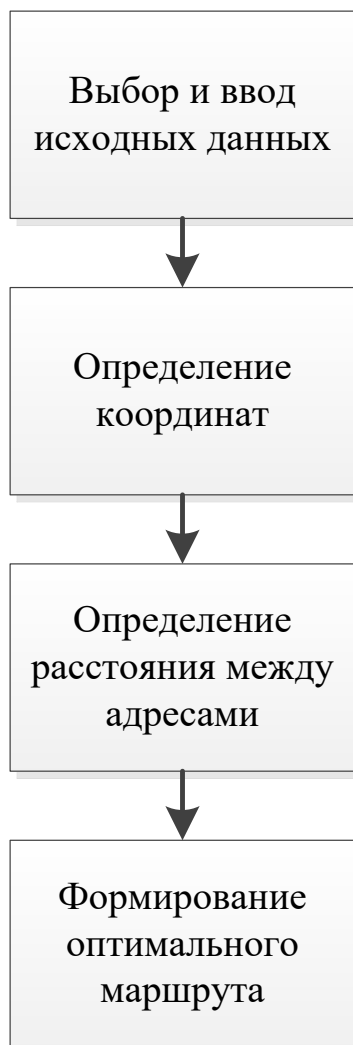


Рисунок 11 - Описание функции

4.4 Описание приложения

Интерфейс системы интуитивно понятен. На основании поступивших заказов диспетчер логистического центра формирует заявку для курьера. Для этого необходимо заполнить форму. Форма имеет 6 столбцов и представляет собой таблицу для заполнения. На первом этапе заполняется столбец носящий имя «Населенный пункт». В выпадающем меню необходимо выбрать нужный

населенный пункт (Рисунок 12). Так как курьерская компания осуществляет свою деятельность не только в городе, но и в районе, то в базу данных помимо самого Усть-Илимска были внесены населенные пункты Усть-Илимского района.

	Населенный пункт	Выбор улицы	№ дома	№ квартиры	Информация о грузе	Действия
1	г. Усть-Илимск	Мира	2	66	Картриджи бшт.	Доставить
2	г. Усть-Илимск	Молодежная	22	1	Заказ и/м "Илимчанка"	Доставить
3	г. Усть-Илимск	Дружбы Народов	44	9	Заказ и/м "Илимчанка"	Доставить
4	г. Усть-Илимск	Карла Маркса	49	14	Ценное письмо	Доставить
5	г. Усть-Илимск	Дружбы Народов	90	35	Цветы	Доставить
6	г. Усть-Илимск	↑				
7	п. Бадарма					
8	п. Высотка					
9	п. Карапчанка					
10	п. Кобляково					
11	п. Невон					
12	п. Седаново					
13	п. Туба					
14						
15						
16						
17						
18						

Рассчитать маршрут

Рисунок 12 – Выбор населенного пункта

На следующем этапе, во втором столбце формы, из выпадающего меню необходимо выбрать нужную улицу.

	Населенный пункт	Выбор улицы	№ дома	№ квартиры	Информация о грузе	Действия
1	г. Усть-Илимск	Мира	2	66	Картриджи бшт.	Доставить
2	г. Усть-Илимск	Молодежная	22	1	Заказ и/м "Илимчанка"	Доставить
3	г. Усть-Илимск	Дружбы Народов	44	9	Заказ и/м "Илимчанка"	Доставить
4	г. Усть-Илимск	Карла Маркса	49	14	Ценное письмо	Доставить
5	г. Усть-Илимск					
6		40 Лет Победы				
7		Белградская				
8		Героев Труда				
9		Димитрова				
10		Дружбы Народов				
11		Интернационалистов				
12		Карла Маркса				
13		Мечтателей				
14						
15						
16						
17						
18						

Рассчитать маршрут

Рисунок 13 - Выбор улицы

Следующим, логическим, шагом работы в программе является выбор номера дома и номера квартиры. Так как курьерская копания работает по принципу «от двери до двери». Как и на предыдущих этапах работы, номер дома и номер нужной квартиры выбирается из выпадающего меню.

	Населенный пункт	Выбор улицы	№ дома	№ квартиры	Информация о грузе	Действия
1	г. Усть-Илимск	Мира	2	66	Картриджи бшт.	Доставить
2	г. Усть-Илимск	Молодежная	22	1	Заказ и/м "Илимчанка"	Доставить
3	г. Усть-Илимск	Дружбы Народов	44	9	Заказ и/м "Илимчанка"	Доставить
4	г. Усть-Илимск	Карла Маркса	49	14	Ценное письмо	Доставить
5	г. Усть-Илимск	Дружбы Народов				
6		83				
7		84				
8		85				
9		86				
10		87				
11		88				
12		89				
13		90				
14						
15						
16						
17						
18						

Рассчитать маршрут

Рисунок 14 - Выбор номера дома

	Населенный пункт	Выбор улицы	№ дома	№ квартиры	Информация о грузе	Действия
1	г. Усть-Илимск	Мира	2	66	Картриджи бшт.	Доставить
2	г. Усть-Илимск	Молодежная	22	1	Заказ и/м "Илимчанка"	Доставить
3	г. Усть-Илимск	Дружбы Народов	44	9	Заказ и/м "Илимчанка"	Доставить
4	г. Усть-Илимск	Карла Маркса	49	14	Ценное письмо	Доставить
5	г. Усть-Илимск	Дружбы Народов	90			
6			33			
7			34			
8			35			
9			36			
10			37			
11			38			
12			39			
13			40			
14						
15						
16						
17						
18						

Расчитать маршрут

Рисунок 15 - Выбор номера квартиры

После заполнения адресных строк необходимо внести информацию о грузе. Спектр услуг курьерской компании довольно широк. В него входит доставка корреспонденции, еды, цветов, товаров интернет магазина, компьютерной перефирии и расходных материалов из сервисного центра. Чтобы курьер легче ориентировался в обилии доставляемого груза, в 5 колонке прописывается информация о грузе для выбранного адресата. Информация вносится сотрудником логистического центра вручную.

Специфика работы курьерской службы такова, что курьер не только доставляет грузы, но и забирает их. Для указания нужного действия существует последняя шестая колонка. Из выпадающего списка необходимо выбрать нужное действие.

После того как вся необходимая информация о всех заказах внесена в форму, оператору логистического центра необходимо нажать на кнопку «Расчитать маршрут».

	Населенный пункт	Выбор улицы	№ дома	№ квартиры	Информация о грузе	Действия
1	г. Усть-Илимск	Мира	2	66	Картриджи бшт.	Доставить
2	г. Усть-Илимск	Молодежная	22	1	Заказ и/м "Илимчанка"	Доставить
3	г. Усть-Илимск	Дружбы Народов	44	9	Заказ и/м "Илимчанка"	Доставить
4	г. Усть-Илимск	Карла Маркса	49	14	Ценное письмо	Доставить
5	г. Усть-Илимск	Дружбы Народов	90	35	Цветы	
6						Забрать
7						Доставить
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						

Рассчитать маршрут

Рисунок 16 - Выбор действия

После нажатия кнопки «Рассчитать маршрут» программа проанализирует выбранные адреса и выдаст список адресов в порядке оптимального маршрута. Далее есть два варианта развития событий. Возможен возврат в форму для редактирования и добавления вновь поступивших данных путем нажатия кнопки «Редактировать». Если редактирование не планируется, то список отправляется на печать.

1	г. Усть-Илимск	Мира	2	66	Картриджи бшт.	Доставить
2	г. Усть-Илимск	Молодежная	22	1	Заказ и/м "Илимчанка"	Доставить
3	г. Усть-Илимск	Дружбы Народов	44	9	Заказ и/м "Илимчанка"	Доставить
4	г. Усть-Илимск	Карла Маркса	49	14	Ценное письмо	Доставить
5	г. Усть-Илимск	Дружбы Народов	90	35	Цветы	Доставить

Редактировать
Печать

Рисунок 17 - Оптимальный маршрут

4.5 Визуализация результатов работы

После внесения оператором адресов в форму для заполнения, программа выбирает данные адреса и представляет их в виде вершин ориентированного графа. А затем по алгоритму Беллмана-Форда рассчитывается оптимальный маршрут.



Рисунок 18 – Построение маршрута

Изм.	Лист	№ документа	Подпись	Дата

ДП-230102.65-1017437ПЗ

Лист

39

ЗАКЛЮЧЕНИЕ

Цель данного проекта - разработка автоматизированной системы расчета оптимального маршрута для курьерской компании.

Для реализации данного проекта были решены следующие задачи:

- изучен процесс доставки грузов;
- проанализирован процесс логистики;
- выбран алгоритм и программа для реализации проекта;
- разработано приложение для расчета оптимального маршрута.

Исходя из цели и поставленных задач, была определена структура данного дипломного проекта. В первой и второй главах производится анализ деятельности курьерской компании и описание услуг. В третьей части производится анализ алгоритмов для создания системы. В четвертой части проведен анализ инструментария, представлена программная реализация для написания приложения и описан результат работы системы.

Результатом решения поставленных задач является создание полноценной автоматизированной информационной системы для расчета оптимального маршрута, которая способна оптимизировать работу компании, повысить скорость доставки грузов и, как следствие - повысить качество курьерских услуг.

Автоматизированная система расчета оптимального маршрута для курьерской компании ориентирована, в первую очередь, на те курьерские компании, которые стремятся оптимизировать свою деятельность, повысить эффективность транспортных перевозок, и повысить свою конкурентоспособность на рынке транспортных услуг.

Во время изучения описанной проблемы и для реализации данного дипломного проекта был проведен мониторинг работы курьерской компании, а так же изучено значительное число литературных и электронных источников, описывающих и изучающих выбранную тему. Весь перечень источников представлен в конце отчета.

					ДП-230102.65-1017437ПЗ	Лист
						40
Изм.	Лист	№ документа	Подпись	Дата		

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бейли Л. Изучаем SQL / Л. Бейли - Санкт-Петербург: Питер, - 2012. - 592 с.
2. Бондарь, А.Г. Microsoft SQL Server 2014 / А.Г. Бондарь. – Санкт-Петербург: БХВ-Петербург, - 2015 – 592 с.
3. Герами, В.Д. Управление транспортными системами. Транспортное обеспечение логистики: учебник / В.Д. Герами. - Москва: Юрайт, - 2014. - 510 с.
4. Еремеева, Л.Э. Транспортная логистика : учебное пособие / Л. Э. Еремеева. - Сыктывкар: Сыктывкарский лесной институт. СЛИ, - 2013. - 260 с.
5. Кадырова, Г. Р. Основы алгоритмизации и программирования : учебное пособие / Г. Р. Кадырова. - Ульяновск : УлГТУ, 2014. - 95 с.
6. Колисниченко, Д. Разработка Web - приложений / Д. Колисниченко. Санкт - Петербург: БХВ - Петербург, - 2013. - 543 с.
7. Никсон, Р. Создаем динамические веб - сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Р. Никсон. - Москва: Питер. - 2015 – 688с.
8. Пауэрс, Д. PHP Solution: Dynamic Web Design Made Easy / Д.Пауэрс. - Москва: Рид Групп, - 2012. - 640 с.
9. Смирнов, С.Н. Практикум по работе с базами данных / С.Н. Смирнов, А.В. Киселев. - Москва: Гелиос АРВ, - 2012 – 160 с.
10. Федорова, Г. Информационные системы 3-е издание / Г. Федорова. - Москва: Академия, - 2013. - 208 с.
11. Щербаков, В.В. Автоматизация бизнес-процессов в логистике: учебник / В.В. Щербаков. - Санкт - Петербург: Питер, - 2016. - 464 с.

										Лист
										41
Изм.	Лист	№ документа	Подпись	Дата	ДП-230102.65-1017437ПЗ					

ПРИЛОЖЕНИЕ А

Техническое задание

Предметная область:

Курьерская компания

Техническое задание:

A1 Введение

Настоящее техническое задание, оформленное в соответствии с ГОСТ 19.201-78, содержит требования к системе тестирования, предназначенной для расчета оптимального маршрута.

A2 Основание для разработки

Основанием для разработки системы тестирования является задание на выпускную квалификационную работу в форме дипломного проекта.

Исполнитель и заказчик

Заказчиком разработки, выполняемой по настоящему ТЗ, является курьерская компания «Миг»

Исполнителем разработки, выполняемой по настоящему ТЗ, является студент группы ЗКИ10-08 Карманов Константин Борисович.

Наименование

Программе, разрабатываемой по настоящему ТЗ, присваивается наименование: Модуль расчета маршрута «courier».

A3 Назначение разработки

Модуль расчета маршрута «courier» предназначен для выполнения следующих функций:

- построения оптимального маршрута;
- предоставлении курьеру информации о грузе;
- предоставлении курьеру информации о необходимом действии;

A4 Требования к программе и программному изделию

Требования к составу

Модуль расчета маршрута «courier» должен состоять графического интерфейса с модулем, выполняющего все требуемые функции.

Требования к функциональным характеристикам

Требования к составу выполняемых функций

Модуль расчета маршрута «courier» должен выполнять следующие функции:

- систематизировать данные;
- составлять оптимальный маршрут.

Требования к составу и параметрам технических средств

Модуль расчета маршрута «courier» должен функционировать с любой операционной системой.

Специальные требования

Требования не предъявляются.

A5 Стадии и этапы разработки

Создание и отработка модуля расчета маршрута «courier» должно производиться по следующим основным этапам:

1-й этап. Разработка модуля.

2-й этап. Испытание модуля.

3-й этап. Подготовка и передача модуля заказчику.

A6 Порядок контроля и приемки

Приемка разработанного модуль расчета маршрута «courier» осуществляется на средствах вычислительной техники Заказчика в объеме, определяемом программой и методикой испытаний модуля расчета маршрута «courier», и в сроки, определенные ведомостью исполнения (календарным планом). Результаты приемки модуля должны быть отражены в техническом акте.

Программа и методика испытаний модуля расчета маршрута «courier» выпускается Исполнителем в соответствии с ГОСТ 19.301-79 («программа и методика испытаний») и должна быть согласована с Заказчиком.

A7 Порядок внесения изменений в техническое задание

Настоящее техническое задание может уточняться и дополняться в процессе создания совместными решениями сторон, подписавших техническое задание, оформленными в виде дополнения к ТЗ в соответствии с требованиями ГОСТ 19.201-78.