**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Институт – Кибернетики

Направление – Информатика и вычислительная техника

пояснительная записка к курсовой работе

**«Мониторинг сети»**

по курсу

«Технология разработки программного обеспечения»

Студент гр. 8ВМ23 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Б.А. Сафронов

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.А. Заикин

**Содержание**

[Введение 3](#_Toc357354809)

[1. Требования к программе 5](#_Toc357354810)

[1.1 User-stories 6](#_Toc357354811)

[1.2 Варианты использования 7](#_Toc357354812)

[2. Анализ 10](#_Toc357354813)

[2.1 Диаграммы классов 10](#_Toc357354814)

[2.2 Диаграммы последовательности 11](#_Toc357354815)

[2.3 Диаграмма деятельности 12](#_Toc357354816)

[3. Проектирование 13](#_Toc357354817)

[3.1 Проектные классы 13](#_Toc357354818)

[3.2 Конечные автоматы 14](#_Toc357354819)

[4. Реализация 15](#_Toc357354820)

[4.1 Тестирование 15](#_Toc357354821)

[4.2 Непрерывная интеграция 16](#_Toc357354822)

[5. Документация 20](#_Toc357354823)

[5.1 Руководство оператора 20](#_Toc357354824)

[6. Заключение 26](#_Toc357354825)

[Список использованных источников 27](#_Toc357354826)

# Введение

Применение современных информационных технологий связано, как правило, с активным использованием сетевого оборудования, а так же программного обеспечения, использующего локальную или глобальную сеть для передачи данных. Сеть является жизненной основой любого бизнеса, поэтому оптимизация [работы сети](http://ru.flukenetworks.com/) жизненно необходима. ИТ-специалисты не могут полагаться на догадки для успешного выполнения такой задачи. Плохо настроенные устройства приводят к потере эффективности работы сети и предприятия в целом, влияя на конечный экономический результат.

Для обнаружения и решения подобных проблем необходимо использовать системы мониторинга сетей и приложений.

Мониторингом сети считается процесс опроса узлов сети на предмет доступности. Мониторинг серверов позволяет своевременно реагировать на события сети.

В современных сетях все активные устройства имеют встроенные агенты для мониторинга – это упрощает задачу своевременного обнаружение неисправностей и узких мест сети. Однако если опрашивать каждый узел – возрастает нагрузка на сеть, поэтому необходимо выделить приоритеты для конкретной инфраструктуры сети. В зависимости от инфраструктуры, положения рабочих серверов и прочих компонентов сети – приоритеты будут расставляться по-разному, однако во всех случаях возникает необходимость постоянного мониторинга пограничных маршрутизаторов (для крупных сетей), серверов авторизации (для доменных сетей) и блоков бесперебойного питания, потому что при перебоях в работе этих устройств возможна полная или частичная потеря работоспособности сети. Кроме проблем стабильности, есть необходимость мониторинга узких мест сетей.

Для реализации системы мониторинга была выбрана платформа .NET. Разработка ASP приложения позволяет добиться доступности статистических данных с любого устройства сети. Кроме того была реализована возможность работы с базой данных узлов с разных контролирующих серверов.

Средой для разработки была выбрана MS Visual Studio 2012. Так же, в ходе выполнения проекта, был использован репозиторий GitHub как система контроля версий и система непрерывной интеграции Jenkins.

# Требования к программе

Данный продукт предназначен для отслеживания доступности удаленных узлов локальной и глобальной сети, используя любое устройство с доступом к сети.

Применение данного продукта возможно как в локальных сетях, для отслеживания доступа к внутренним узлам сети, так и в глобальных, для отслеживания доступности интернет узлов.

В ходе работы, программа проверяет доступность заданных узлов сети, путем отправки icmp пакетов. В зависимости от доступности узла в базу данных заносится соответствующая информация.

Кроме возможности добавления отдельных узлов, предусмотрена функция сканирования подсети. В ходе работы сканера, проверяются все узлы, попадающие в заданный диапазон. Найденные узлы помещаются в общую базу данных.

## User-stories

Как пользователь, я хочу наблюдать за доступностью узлов.

* На странице мониторинга доступна таблица, включающая состояние всех опрашиваемых узлов;
* На странице узлов доступна таблица, включающая все параметры узлов и время последнего опроса узла.

Как пользователь, я хочу добавить новый узел для опроса.

* На странице добавления узла необходимо задать ip адрес узла или его имя и нажать кнопку «Добавить».

Как пользователь, я хочу добавить диапазон узлов для опроса.

* На странице сканирования диапазона необходимо указать начальный и конечный ip адреса. После нахождения доступного узла в диапазоне, он будет добавлен в общую базу данных.

Как пользователь, я хочу разделить устройства на группы.

* При добавлении нового узла или задании параметров сканирования, необходимо заполнить поле «Группа».
* Для уже добавленных узлов, на странице списка узлов сети, возможно изменение параметра «группа» для каждого узла.

Как пользователь, я хочу сохранить текущую базу данных и начать работу с новой (или использованной ранее).

* На странице настроек, необходимо в поле «База данных» ввести название новой (или существующей) базы данных.

## Варианты использования

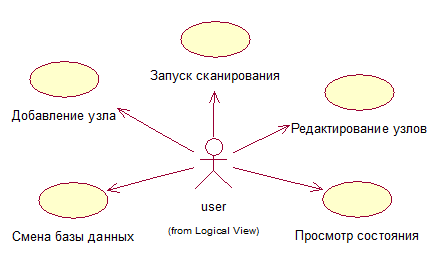


Рисунок . Диаграмма вариантов использования

**Добавление узла**

Актер: пользователь

Предусловия: наличие доступа к веб интерфейсу

Основной поток:

1. пользователь переходит на страницу добавления узла
2. пользователь заполняет необходимые поля
3. пользователь нажимает кнопку «Добавить»

Постусловие: узел добавлен в базу

**Запуск сканирования**

Актер: пользователь

Предусловия: наличие доступа к веб интерфейсу

Основной поток:

1. пользователь переходит на страницу сканирования
2. пользователь заполняет необходимые поля
3. пользователь нажимает кнопку «Начать сканирование»

Постусловие: параметры сканирования переданы для обработки

**Смена базы данных**

Актер: пользователь

Предусловия: наличие доступа к веб интерфейсу

Основной поток:

1. пользователь переходит на страницу настроек
2. пользователь заполняет поле адреса базы данных
3. пользователь нажимает кнопку «Применить»

Постусловие: адрес рабочей базы данных изменен

**Редактирование узлов**

Актер: пользователь

Предусловия: наличие доступа к веб интерфейсу

Основной поток:

1. пользователь переходит на страницу списка узлов
2. пользователь выбирает узел для редактирования и нажимает кнопку «Изменить»
3. пользователь вводит новые параметры для выбранного узла
4. пользователь нажимает кнопку «Обновить»

Постусловие: параметры узла обновлены

**Просмотр состояния**

Актер: пользователь

Предусловия: наличие доступа к веб интерфейсу

Основной поток: пользователь переходит на страницу «Мониторинг»

Постусловие: пользователь видит текущее состояние узлов

# Анализ

## Диаграммы классов

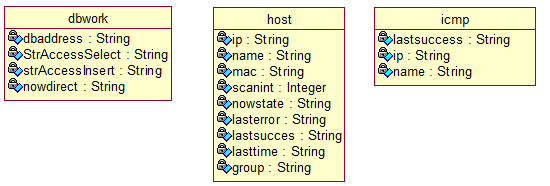


Рисунок . Диаграмма классов

На диаграмме классов можно увидеть все атрибуты класса. Для выполнения задачи мониторинга необходимо использовать 3 класса.

**Класс dbwork** используется для обращения к базе данных. Имеет поля, содержащие адрес базы данных и строки для запроса или записи данных.

**Класс host** используетсядля структуризации параметров узла и последующей работы с ним. Содержит поля ip адреса, имени, мак адреса, интервала сканирования, текущего состояния, последней ошибки, последнего успешного выполнения, последнего обращения и группы.

**Класс icmp** используется для опроса узлов. Содержит поля ip адрес, имя и время последнего удачного опроса.

## Диаграммы последовательности

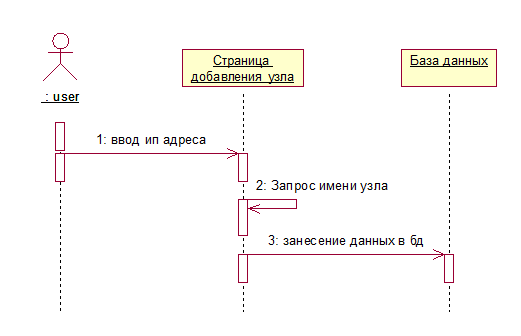


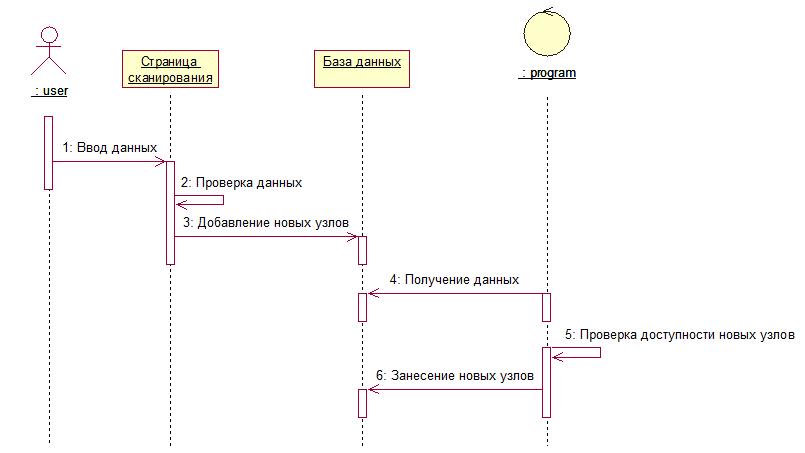
Рисунок . Диаграмма последовательностей для добавления узла

Рисунок . Диаграмма последовательностей для процесса сканирования

Диаграмма последовательности показывает последовательность выполнения заданного действия.

## Диаграмма деятельности

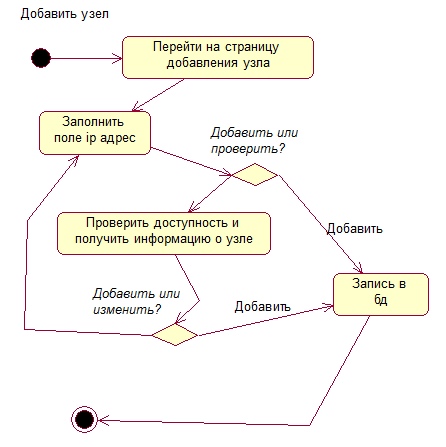


Рисунок . Диаграмма деятельности для варианта «Добавить узел»

Диаграмма деятельности показывает все этапы выполнения заданного варианта использования.

# Проектирование

## Проектные классы

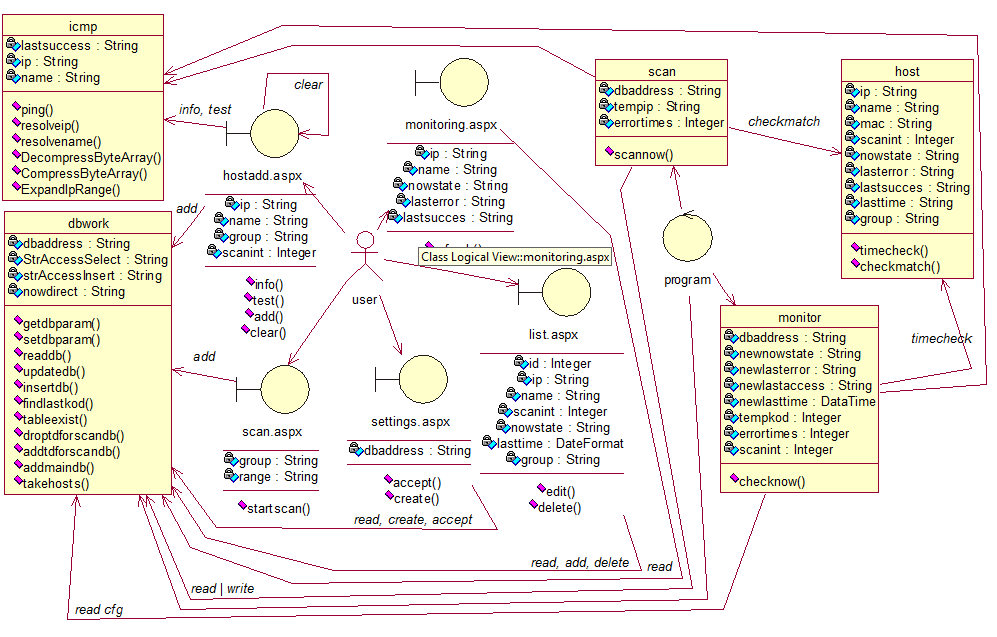


Рисунок . Диаграмма классов

На диаграмме показана структура системы, включая атрибуты и методы классов.

## Конечные автоматы

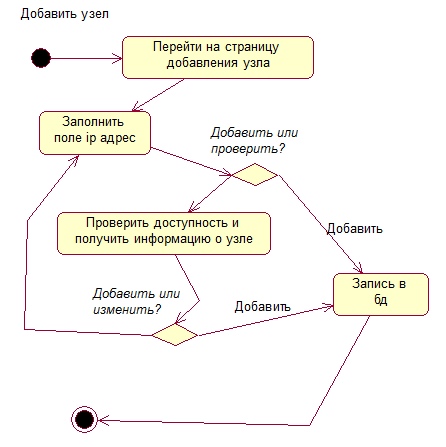


Рисунок . Диаграмма состояний для варианта «Добавить узел»

Диаграмма состояний показывает переходы между состояниями при выполнении процесса добавления нового узла в базу данных.

# Реализация

## Тестирование

Для тестирования использовалась стандартная возможность среды разработки. В процессе написания продукта, добавлялись необходимые тесты.

Пример реализации UnitTest для проверки функции ping:

[TestMethod]

public void IcmpPingTest()

{

IPAddress ip = IPAddress.Parse("127.0.0.1");

newicmp.ping(ip);

}

[TestMethod]

[ExpectedException(typeof(System.FormatException))]

public void IcmpPingExceptionTest()

{

IPAddress ip = IPAddress.Parse("");

newicmp.ping(ip);

}

Кроме ручного запуска тестов, был реализован запуск тестов во время автоматического построения проекта в среде Jankins.

## Непрерывная интеграция

В ходе написания проекта, была реализована возможность автоматической сборки и запуска тестов проекта. Для этих целей использовался продукт Jankins, репозиторий github и конфигурационный файл автоматической сборки.

В файле задается цель, в данном случае запуск тестов. Для каждого этапа устанавливаются зависимости, в нашем случае запуск тестов зависит от компиляции проекта. Это означает, что перед запуском тестов, проект будет скомпилирован. Кроме последовательности выполнения, в файле прописаны общие параметры сборки и версия процессора.

Пример конфигурационного файла:

<?xml version="1.0" encoding="utf-8" ?>

DefaultTargets="Test">

<ItemGroup>

<BuildArtifactsDir Include="BuildArtifacts\" />

<SolutionFile Include="MS.sln" />

</ItemGroup>

<PropertyGroup>

<Configuration Condition=" '$(Configuration)' == '' ">Release</Configuration>

<BuildPlatform Condition=" '$(BuildPlatform)' == '' ">Any CPU</BuildPlatform>

</PropertyGroup>

<Target Name="Init" DependsOnTargets="Clean">

<MakeDir Directories="@(BuildArtifactsDir)" />

</Target>

<Target Name="Clean">

<RemoveDir Directories="@(BuildArtifactsDir)" />

</Target>

<Target Name="Compile" DependsOnTargets="Init">

<MSBuild Projects="@(SolutionFile)" Targets="Rebuild"

Properties="OutDir=%(BuildArtifactsDir.FullPath);Configuration=$(Configuration);Platform=$(BuildPlatform)" />

</Target>

<Target Name="Test" DependsOnTargets="Compile">

<PropertyGroup><TestSuccess>1</TestSuccess> </PropertyGroup>

<Exec Command='" MSTest.exe" "/testcontainer:@(BuildArtifactsDir)\unittestms.dll"' >

<Output TaskParameter="ExitCode" PropertyName="TestSuccess"/>

</Exec>

</Target>

В ходе создания новых промежуточных версий проекта, из репозитория копировались файлы программного кода на сервер автоматической сборки. После чего проект, используя конфигурационный файл, компилировался, и запускались необходимые тесты. Пользователь уведомлялся о корректной работе приложения или об ошибках при сборке.

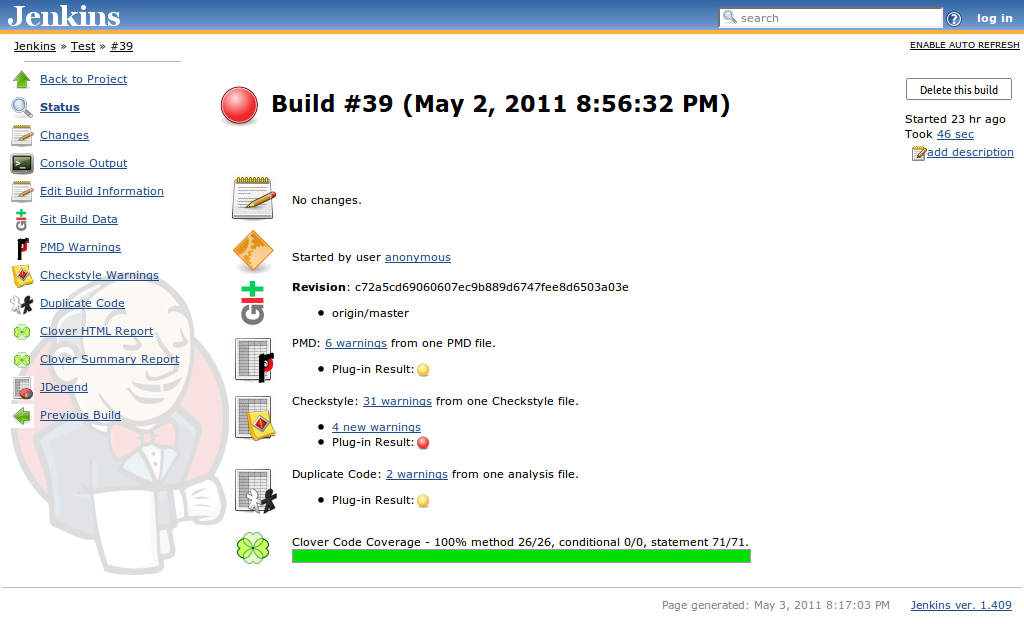


Рисунок . Отчет о попытке построения проекта

Развертывание

Для запуска веб приложения, потребуется среда, поддерживающая работу ASP приложений. При развертывании приложения, необходимо использовать .NET Framework версии не ниже 4.0, а так же AccessDatabaseEngine (OleDb). В зависимости от разрядности вашего компьютера, рекомендуется устанавливать соответствующую версию продукта.

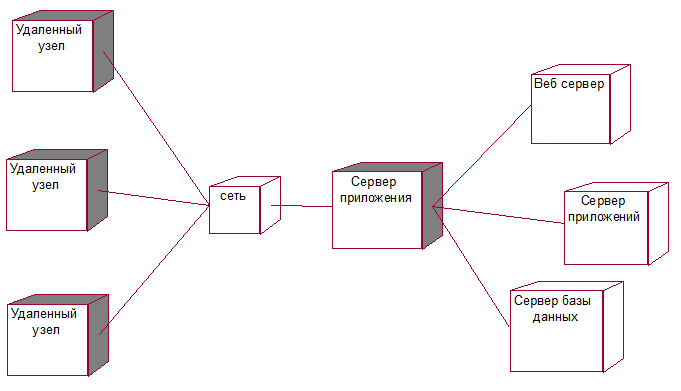


Рисунок . Диаграмма развертывания

Диаграмма развертывания представляет собой модель физического размещения приложения. Модель включает в себя физические узлы, необходимые программные компоненты и тип соединения между устройствами.

Приложение было развернуто на сервере доступа VPN частной сети. Сервер имеет 64 битную архитектуру, соответственно для реализации работы приложения был использован AccessDatabaseEngine\_x64. Так как сервер уже использовался в качестве веб сервера (MS IIS3) был добавлен новый сайт и указан путь к файлам веб приложения.

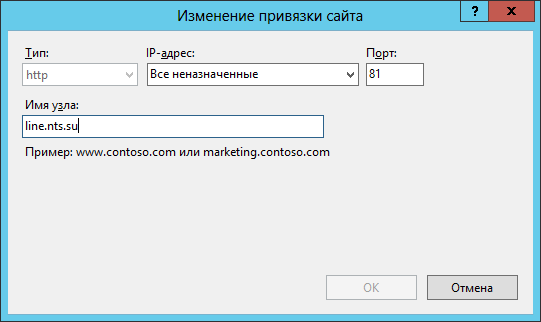


Рисунок . Развертывание веб-приложения.

# Документация

## Руководство оператора

Для начала работы с программой, необходимо запустить любой веб браузер и ввести имя (или ip адрес) хоста, на котором произведена инсталляция.

Если это первый запуск и необходимо настроить базу данных – необходимо перейти на вкладку «Настройки» для ввода названия базы данных. По умолчанию база данных называется hosts.accdb.

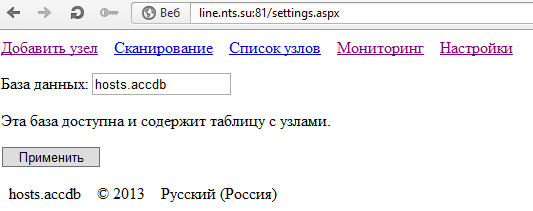


Рисунок . Страница настроек

Для изменения рабочей базы данных, введите в строку «База данных» название новой или существующей БД. Если база данных уже существует, конфигурационный файл будет изменен для работы с существующей базой данных, иначе будет предложено создать новую базу данных.

После инициализации базы данных, пользователь будет перемещен на страницу «мониторинг».

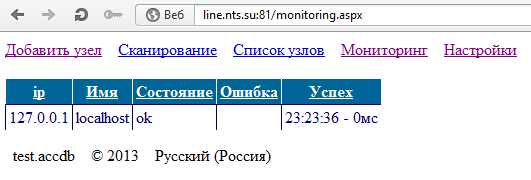


Рисунок . Инициализация новой базы данных

Если это новая база данных, будет доступен только один локальный адрес.

Для добавления нового удаленного узла, необходимо перейти на вкладку «Добавить узел».

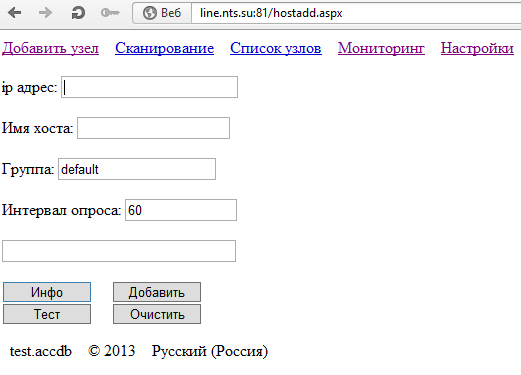


Рисунок . Странице добавления нового узла

Чтобы добавить узел, необходимо написать его ip адрес или имя хоста в соответствующее поле и нажать кнопку «Инфо».

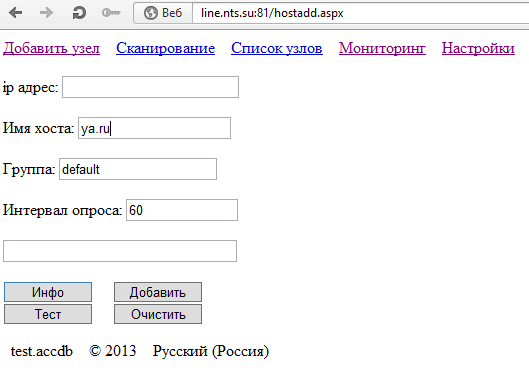


Рисунок . Ввод имени узла

Приложение найдет записи DNS имен для заданных параметров и вернет полученные значения на форму. Для добавления узла в базу опроса, необходимо нажать кнопку «Добавить».

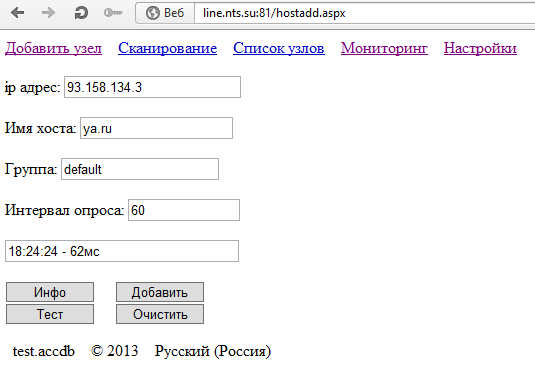


Рисунок . Найдены данные для узла ya.ru

Для добавления диапазона адресов, необходимо перейти на вкладку «Сканирование».

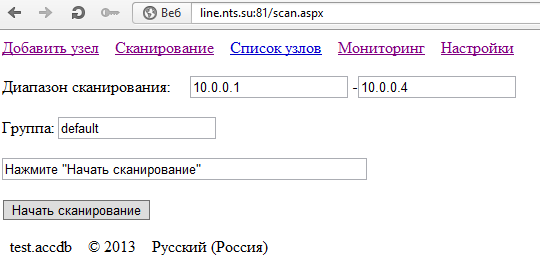


Рисунок . Страница сканирования диапазона адресов

Для начала сканирования, необходимо ввести начальный и конечный адрес и нажать кнопку «Начать сканирование», после чего можно покинуть страницу – сканирование доступных узлов и их добавление в общую базу произойдет автоматически.

Для просмотра и редактирования добавленных узлов, необходимо перейти на вкладку «Список узлов».

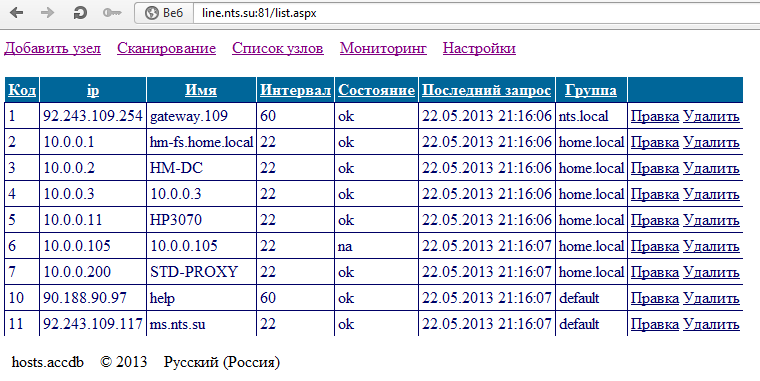


Рисунок . Список добавленных узлов

На этой странице можно получить информацию об узле сети, а так же интервал опроса, последнее состояние и время последнего опроса.

Для редактирования записи, необходимо нажать «Правка» в правой части таблицы.

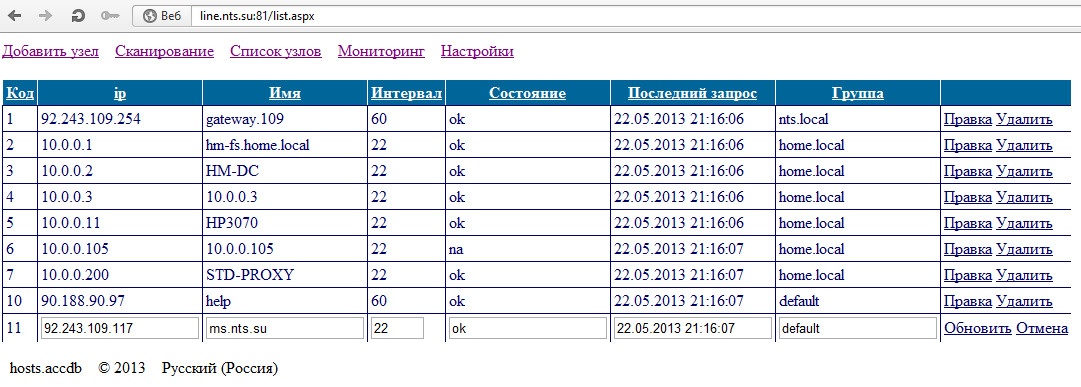


Рисунок . Редактирования информации о узле сети

После ввода необходимых поправок и нажатия кнопки «Обновить», запись будет обновлена в текущей базе данных. Для отмены редактирования, необходимо нажать кнопку «Отмена».

Для удаления записи из списка, необходимо нажать кнопку «Удалить».

Для просмотра текущего состояния, времени последнего сбоя или последнего успешного опроса узла, необходимо перейти на страницу «Мониторинг».

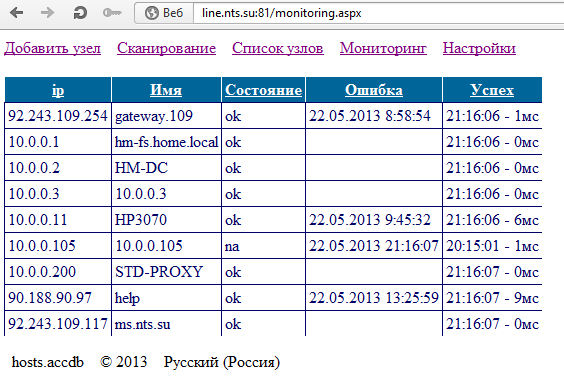


Рисунок . Страница состояния узлов

Для окончания сеанса работы, необходимо закрыть окно браузера.

# Заключение

В ходе выполнения проекта, была создана программа мониторинга узлов сети. Согласно требованиям, доступ к программе возможен с любого устройства, имеющего доступ к сети и веб браузер. Программа использует .NET технологии, благодаря чему исполняемый файл имеет минимальный размер, а веб часть может быть запущена на любом компьютере с необходимым программным обеспечением.

Для выполнения работы были изучены материалы по теме «Практический объектно-ориентированный анализ и проектирование», а так же технологии создания конфигурационных файлов для систем автоматического построения и проверки проектов. Так же были использованы системы контроля версий.

# Список использованных источников

1. Арлоу Д., Нейштадт И. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование, 2­е издание. – Пер. с англ. – СПб: Символ­Плюс, 2007. – 624 с., ил.