# Требования к программе

## Бизнес требования

ПО предназначено для предоставления информации о русском писателе Гоголе Н.В.

## Пользовательские требования

* Просмотр биографии писателя
* Просмотр фотографий связанных с писателем
* Просмотр произведений писателя
* Скачиваний произведений писателя в формате fb2.zip
* Удобная навигация по всем вышеописанным пунктам

## Функциональные требования

ПО должно быть написано на HTML, JavaScript и использовать библиотеки: AngularJS, Bootstrap, JQuery.

Интерфейс сайта должен быть максимально простым, и иметь в большинстве своем светлые тона.

Все страницы сайта должны содержать навигацию по страницам со следующими пунктами:

* Главная
* Биография
* Фотогалерея
* Произведения

На главной страницы сайта должны быть:

* Фотографии писателя
* Ссылка на биографию писателя
* Ссылки на 3 книги писателя с небольшим вступлением

Произведения на главной странице должны меняться при повторной загрузке страницы.

Биография писателя должна быть разбита на ключевые моменты с возможностью переходить на них из навигации.

ПО должно использовать удобную галерею для просмотра фотографий с возможностями:

* Перелистывание фотографий впереди и назад
* Полноэкранный просмотр фотографий
* Автоматическое перелистывание фотографий при загрузке страницы
* Разный порядок просмотра фотографий при повторной загрузке страницы
* Просмотр миниатюр фотографий

Страница с произведениями писателя должна содержать не менее трех произведений с обложками книг и возможностью прочтения и скачивания книг.

Произведения не имеющие книг должны располагаться ниже со ссылками на статьи в Википедии.

## Требования к ПО

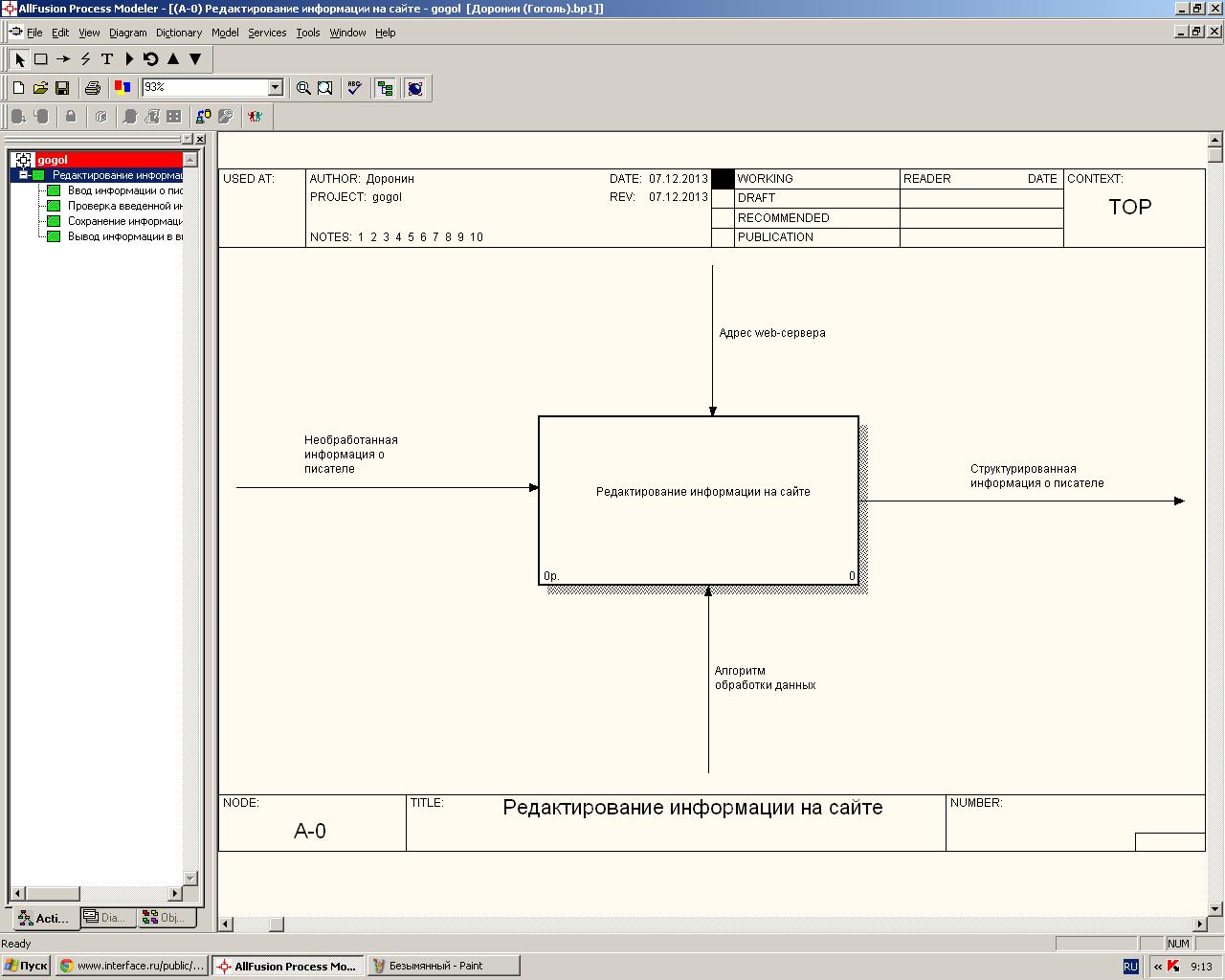
Для корректного отображения сайта, требуется браузер:

* Google Chrome 31
* Opera 17
* Firefox 24
* Internet Explorer 10

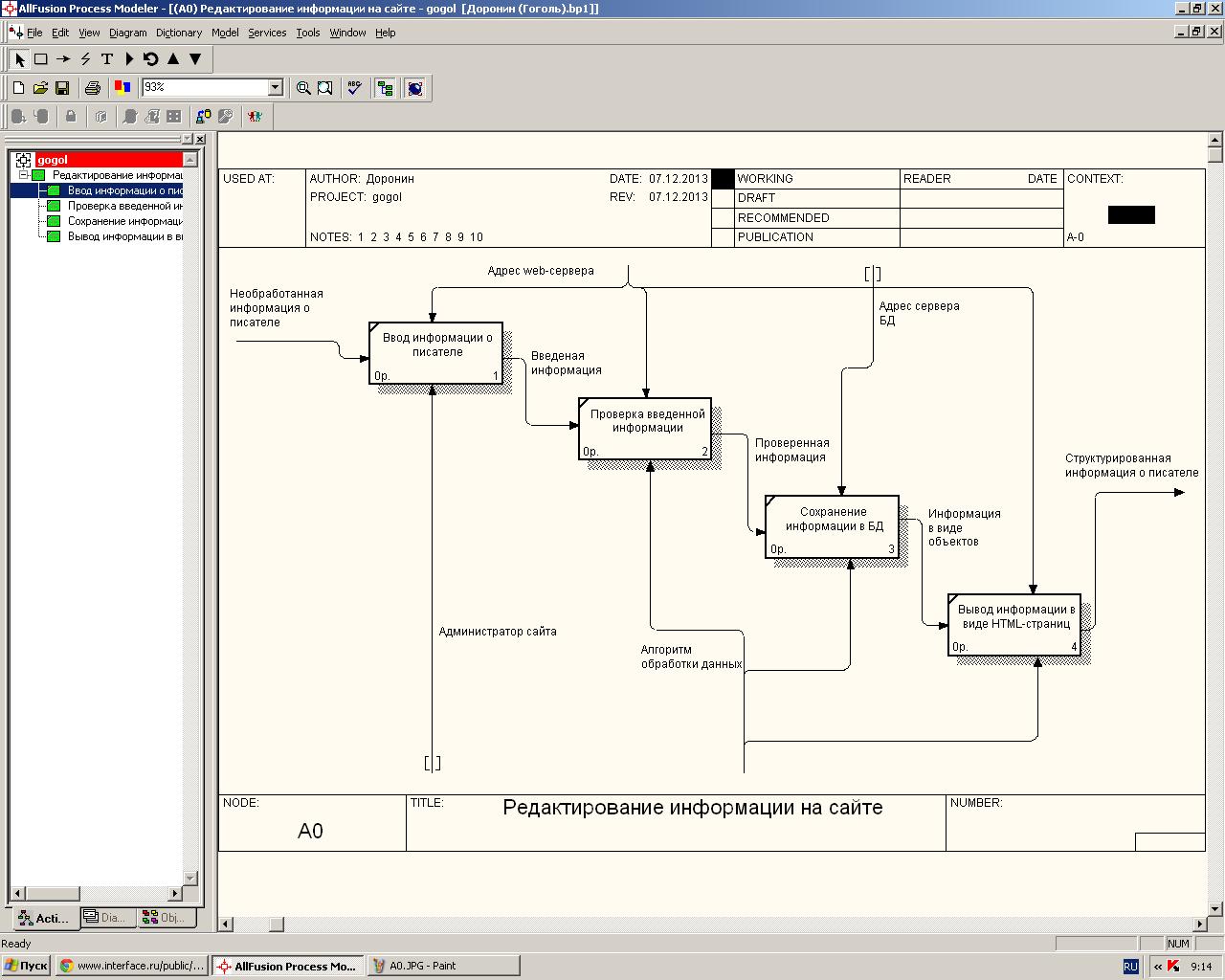
Для чтения книг в формате fb2, рекомендована программа FB2Reader.

# IDEF0

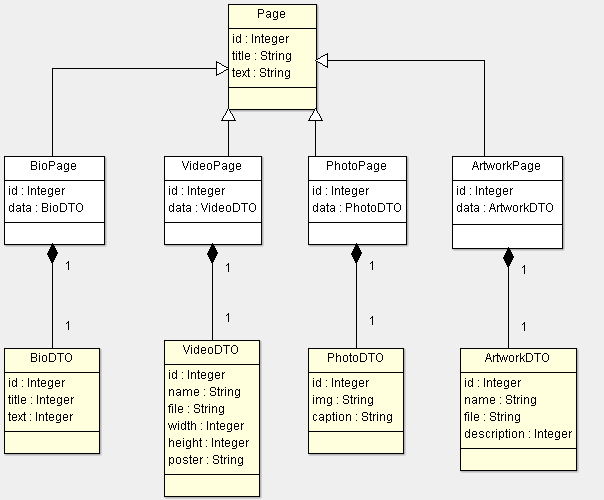
## A-0



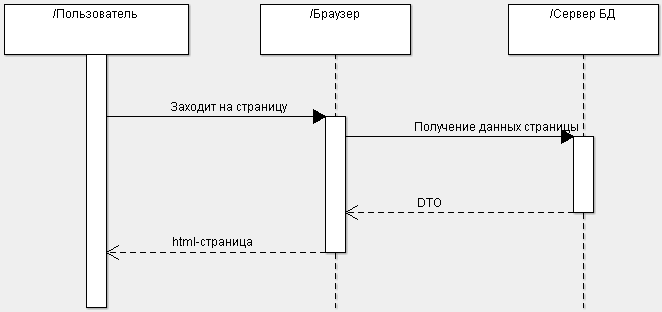
## Декомпозиция A0



# UML-диаграмма классов



# Диаграмма последовательностей



# Тестирование

## Белый ящик

describe(**'App controllers'**, **function**() {

describe(**'MainCtrl'**, **function**() {

**var** rootScope, scope, ctrl, $httpBackend;

beforeEach(module(**'myApp'**));

beforeEach(inject(**function**(\_$httpBackend\_, $rootScope, $controller) {

$httpBackend = \_$httpBackend\_;

$httpBackend.expectGET(**'data/artwork-list.json'**).

respond({popular: [{name: **'Book 1'**}, {name: **'Book 2'**}, {name: **'Book 3'**}, {name: **'Book 4'**}]});

rootScope = {changeActiveNav: **function**(){}};

scope = $rootScope.$new();

ctrl = $controller(**'MainCtrl'**, {$scope: scope, $rootScope: rootScope});

}));

it(**'Должен создать "data" модель с 3-мя книгами'**, **function**() {

expect(scope.data).toBeUndefined();

$httpBackend.flush();

expect(3).toBe(scope.data.length);

});

});

});

## Черный ящик

describe(**'Gogol App'**, **function**() {

describe(**'Тестирование шаблона'**, **function**() {

it(**'Должен перебросить с index.html на index.html#/main'**, **function**() {

browser().navigateTo(**'../../index.html'**);

expect(browser().location().url()).toBe(**'/main'**);

});

it(**'Должен проверить копирайт'**, **function**() {

browser().navigateTo(**'../../index.html'**);

expect(binding(**'copyright'**)).toBe(**'Анна Доронина 2013'**);

expect(element(**'div.footer p'**).text()).toBe(**'© Анна Доронина 2013'**);

});

});

});

# Руководство программиста

1. Условия применения программы
   1. Требования к техническим средствам

* Требования к техническим средствам ИС «Сайт о Гоголе».
* Для работы сайта необходим сервера с конфигурацией не ниже:
* Intel Pentium IV 3Ghz
* 1GB RAM
* 40GB HDD
* Интернет канал со скоростью не ниже 10Mb/s
  1. Требования к общему программному обеспечению (ОПО)

Требования к общему программному обеспечению (ОПО), необходимому для ИС «Сайт о Гоголе», представлены в (Таблица 1).

Таблица 1. Требования к ОПО

|  |  |
| --- | --- |
| Наименование | Количество |
| Серверная ОС | 1 |
| PostgreSQL 8.4 | 1 |
| Программные зависимости | 1 |

1. Характеристика программы

Система рассчитана на круглосуточный режим работы в течение 365 дней в году.

Средствами контроля и мониторинга являются автоматизированные встроенные в платформу программные компоненты, которые самостоятельно выполняют следующие действия в случае ошибки

1. Система автоматически перезапускает себя в случае некритической ошибки

2. Посылает администратору сообщение об ошибке в письме на электронную почту

3. Записывают ошибку в журнал, который находится в файловой системе в файле /var/log/messages

4. В случае невозможности совершить вышеуказанные действия, система выведет администратору ошибки прямо в браузер с конкретными данными о том, где, почему и что произошло.

1. Обращение к программе

Внешнее управление содержимым системы осуществляется через веб-интерфейс, описанный в документе «руководство пользователя».

Внутреннее управление сайтом происходит на сервере.

Структура проекта находится в папке /var/www/projects/gogol и является полноценным набором файлов и программных компонентов для управления логикой, функциональной частью системы и ее работой.

Процесс запуска сайта

Чтобы запустить портал программисту необходимо выполнить ряд шагов:

1. Авторизоваться в терминале сервера под ролью root

2. Перейти в папку проекта с помощью команды cd /var/www/projects/gogol

3. Запустить сервер приложения: scripts/RunServer

Сайт должен запуститься на 80-ом порту, после вывода следующих строчек:

=> Http Server running at <http://localhost:80/>

Входящие запросы будут приниматься по любым интернет-адресам, которые адресуют пользователя на данный сервер.

Проект состоит из следующих папок:

**js** – исходные коды контроллеров. Каждый контроллер управляет одной или несколькими сущностями, содержа в себе набор действий. Каждое действие формирует один URL-адрес

**partials** – исходные коды представлений сайта

**config** – конфигурационные файлы проекта

**db** – директория, содержащая файлы, необходимые для работы с СУБД

**doc** – директория, зарезервированная под хранение внутренней документации

**scripts** – набор автоматизирующих скриптов

**test** – директория для хранения тестов, как Unit, так и интеграционных

**tmp** – директория для хранения временных файлов

1. Входные и выходные данные

В данном разделе приводится описание организации используемой входной и выходной информации.

Внесение изменений в инсталляцию

Внесение изменений в работающую версию будет принято после перезапуска сервера. Перезапуск сервера может быть осуществлен как с остановкой HTTP-сервера и прерыванием соединений, так и без них.

Полная перезагрузка HTTP-сервера nginx в среде Linux выполняется с помощью команды /etc/init.d/nginx restart. Для мягкой перегрузки программной среды, необходимо выполнить команду touch ~/tmp/restart.sh.

Полная перезагрузка требуется в крайне редких случаях и в условиях эксплуатации к использованию рекомендована мягкая перегрузка.

1. Сообщения

В ходе выполнения программы, программисту выдаются сообщения. Здесь приводится описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

**Возможные ошибки, которые могут появиться при заходе на веб-страницу сайта:**

*Error 505*

Решение: мягкая перезагрузка http-сервера

**Возможные ошибки в журнале ошибок (файл /var/log/messages в файловой системе сервера):**

*[warn] mod\_fcgid: Read data error, fastcgi server has close connection*

*[error] [client xxx.xxx.xxx.xxx] Premature end of script headers: dispatch.fcgi*

Решение: мягкая перезагрузка http-сервера

Возможные ошибки в любом из мест:

*PGSQL: Connection pool ended*

*IO: Too much open files*

Решение: необходимо ввести в терминале сервера команду fs.filemax=16452

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

|  |  |
| --- | --- |
| ППО | Прикладное программное обеспечение |
| ИС | Информационная система «Единый портал внешнеэкономической информации Минэкономразвития России» |
| СУБД | Система Управления Базами Данных |

# Руководство пользователя