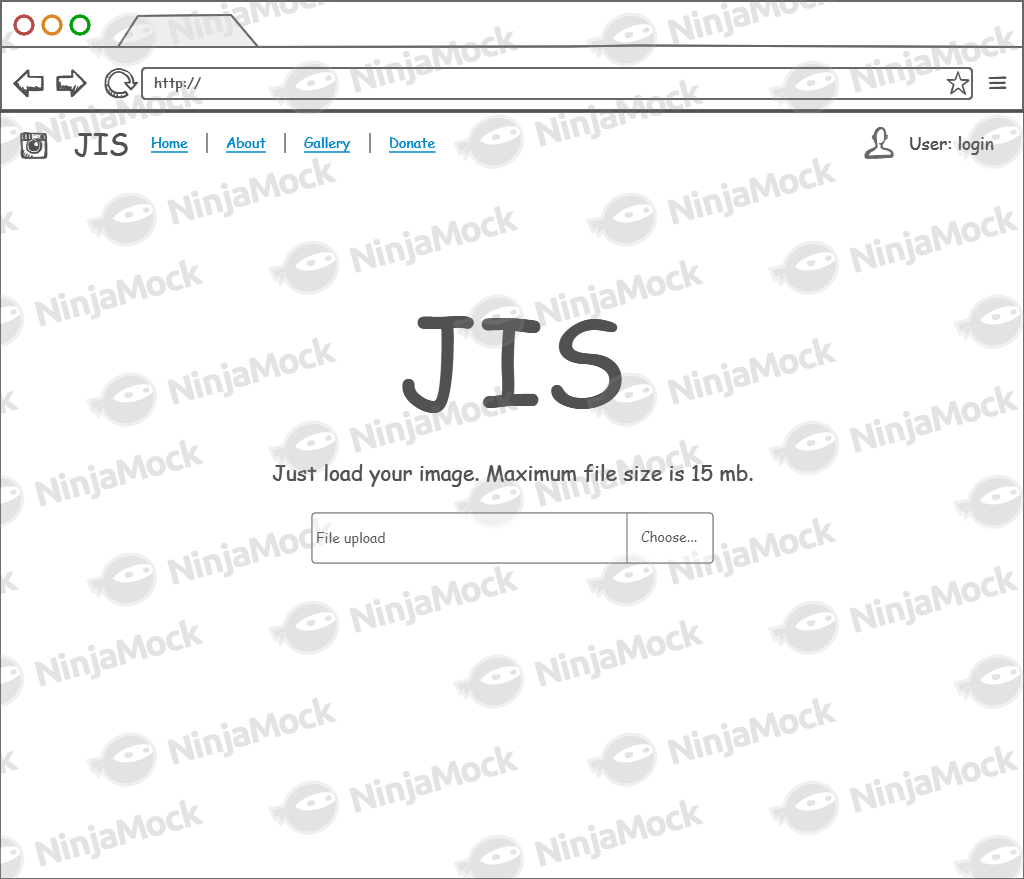
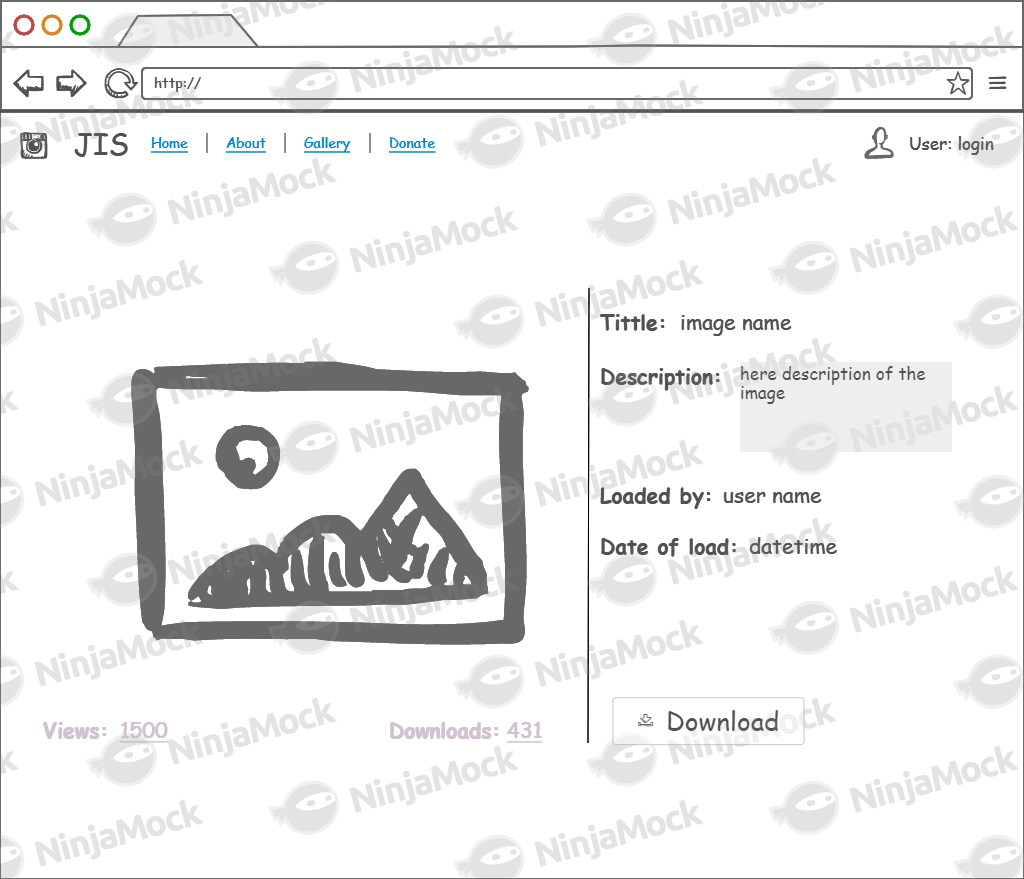
**I. Список форм приложения**

1. Страница загрузки изображения. По совместительству – стартовая страница сайта. Позволяет загрузить изображение в сервис.

Изображение 1

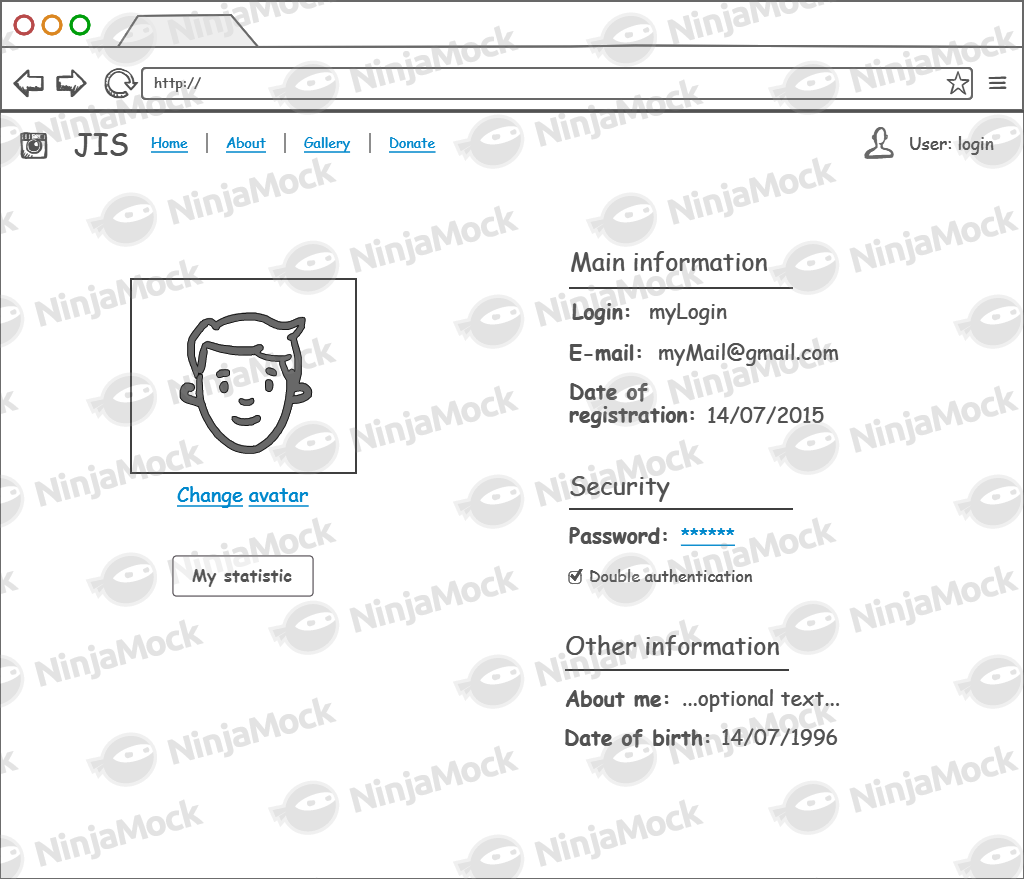
2. Страница просмотра изображения. Позволяет просматривать конкретное изображение к высоком разрешении. Так же содержит дополнительную информацию по изображению, в том числе статистику по кол-ву просмотров/скачиваний. Присутствует функция скачивания изображения.

Изображение 2

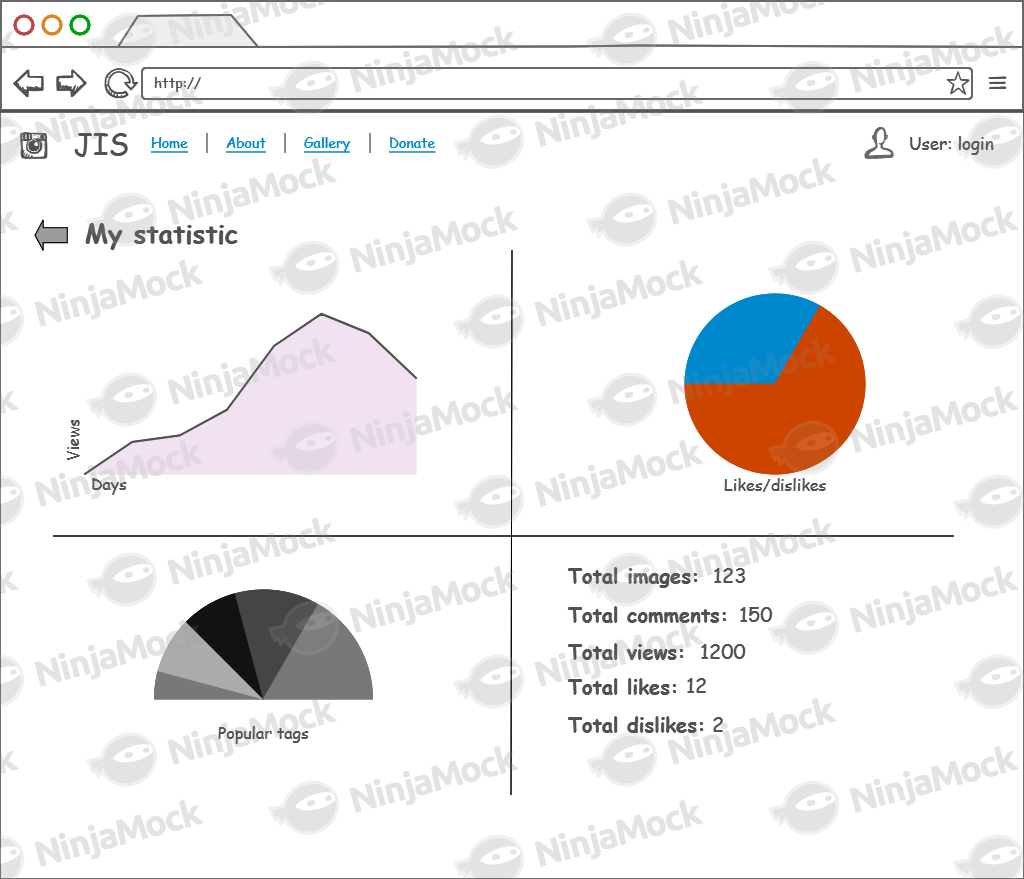
3. Галерея. Страница, содержащая все доступные изображения. Отображаются в виде миниатюр. По клику можно перейти к странице подробного показа изображения (п. 2).

  
Изображение 3

4. Страница профиля пользователя. Показывает подробную информацию о вашем пользователе и предоставляет возможности по ее редактированию.

Изображение 4

5. Страница статистики пользователя. Показывает статистику по: кол-ва просмотров изображений, соотношение лайков/дизлайков изображений, кол-во загруженных изображений по тегам и пр.

Изображение 5

6. Окно редактирование изображения. Модальное окно, позволяющее изменить основную информацию по изображению (название, описание).

Изображение 6

7. Страница авторизации. Позволяет войти зарегистрированному пользователю в систему.

  
Изображение 7

**II. API системы**

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Входные данные | Выходные данные |
| LoadImage | Загрузка изображения из файла | String: imagePath | ImageModel: model |
| SaveImage | Преобразование изображения в бинарный формат и сохранение в БД | ImageModel: model | - |
| GetImage | Подтягивание из БД нужного изображения, и преобразование его из двоичного вида. | Int: imageId | ImageModel: model |
| EditImage | Изменение параметров загруженного изображения | ImageModel: model | - |
| GetImagesByParams | Получение коллекции отфильтрованных изображений | Param[]: params | ImageModel[]: model |
| GetUser | Получение информации по пользователю из БД | Int: userId | User: user |
| Authorization | Попытка авторизации пользователя | String: login, string: password | Bool: authResult |
| GetStatistic | Построение статистики по конкретному пользователю | Int: userId | Statistic: stats |
| Registration | Регистрация нового пользователя | User: user | Bool: regResult |
| AddComment | Добавления комментария к изображению | String: comment, int: imageId | - |

**III. Иерархическая структура работ:**

1. Разработка технического задания:

1.1. Определение требований к проекту  
 1.2. Планирование сроков разработки  
 1.3. Планирование бюджета проекта

2. Проектирование и разработка БД:

2.1. Создание основных сущностей-таблиц  
 2.2. Определение constraint’ов  
 2.3. Создание хранимых процедур/функций для реализации основного функционала.

3. Проектирование и разработка DAL уровня:

3.1. Выбор нужной технологии для мапинга данных между БД и моделями  
 3.2. Создание основных моделей данных  
 3.3. Проектирование и реализация репозиториев

4. Реализация бизнес-логики:

4.1. Создание модуля авторизации и поддержка ролей пользователей.  
 4.2. Создание модуля подсчета статистики  
 4.3. Создание сервиса файлового менеждера

5. Проектирование и реализация UI:

5.1. Проектирование интерфейса основных страниц  
 5.2. Создание контроллеров  
 5.3. Создание view-моделей  
 5.4. Верстка страниц