**ImageCMS WishList**

Сompatible with “[Мodule documentation standard](http://#)”

ImageCMS

Last updated May 24, 2013.

[About](#h.oja6lst0hlco)

[Link’s](#h.inm1m7qoc157)

[Installation](#h.glvfkx3596bi)

[Module configuration](#h.xe0yoeptf38q)

[Module Administration](#h.vqngxvdtkgxe)

[Site Integration](#h.xudy59rrpvkx)

[Вывод кнопок добавления в Список Желания](#h.zdy97xrpmhpz)

[Также после установки модуля будут доступны следующие возможности:](#h.ryjqxygzck41)

[Use case](#h.i2zrbekkblkz)

[Use Case: #1](#h.29wwg8r1yesg)

[Use Case: #2](#h.u35bwc1rlyje)

[Use Case: #3](#h.u35bwc1rlyje)

[Use Case: #4](#h.u35bwc1rlyje)

[Use Case: #5](#h.u35bwc1rlyje)

[Use Case: #6](#h.u35bwc1rlyje)

[Use Case: #7](#h.u35bwc1rlyje)

[Use Case 1 - 7 в виде схемы](#h.u35bwc1rlyje)

[Development info.](#h.pr3gacrmsf75)

[Класс ParentWishlist](#h.g4chxvjkqyww)

[Properties](#h.udozrs2v40rv)

[Public methods](#h.1y8d88qrhpm2)

[API’s](#h.6sp8xmcqsxjh)

[Models](#h.7wab6c33wunn)

[FAQ](#h.adoeks5t3no)

# **About**

Wishlist - модуль который который позволяет пользователям магазина работать со Списками Желания. Список Желаний является списком товаров, в который попадают те наименования, который пользователь желает приобрести, но это действие откладывается по некоторым причинам. Пользователь, создавший список желаний также обладает возможностью публично заявить об интересе к определенным наименованиям, поделившись ссылкой через социальные сети либо электронную почту.

# **Link’s**

# **Installation**

Для начала работы с модулем его нужно [установить](http://wiki.imagecms.net/Модули).

# 

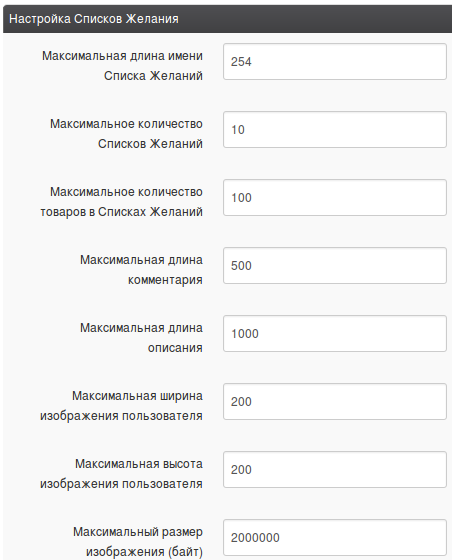
# 

# **Module configuration**

В настройках модуля (Модули->Все модули->wishlist), можно указать ограничения для:

* длины имени списка желания;
* количества списков;
* количества товаров в списках;
* длины комментария;
* длины описания;
* ширины изображения пользователя;
* высоты изображения пользователя;
* размера изображения(байт).

Окно настроек Списка Желаний:



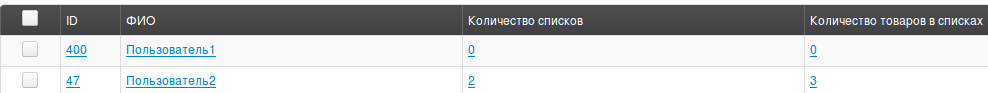
# **Module Administration**

В администрировании модулем (Модули -> Все модули -> wishlist) есть возможности:

* просмотреть і управлять информацией пользователей которые имеют Списки желаний;
  + редактировать им’я, день рождения, описание;
  + удалять/загружать картинку.
* просмотреть і управлять их списками;
  + удалять;
  + редактировать.
* просмотреть і управлять наименованиями в списках;
  + удалять;
  + перемещать между списками;
  + редактировать комментарии;
* создавать новые списки.

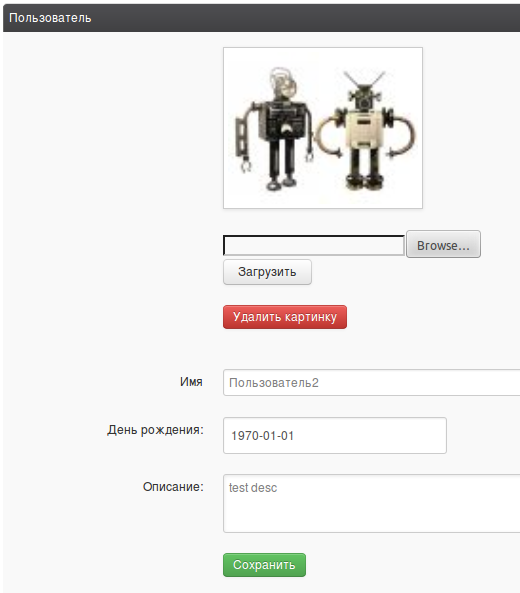
**Страница пользователей:**

На “Странице пользователей” вводится список пользователей которые создали списки желаний. Наводится их имя, количество списков и количество товаров в списках.

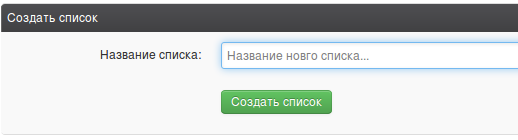


**Управления информацией пользователя:**

На етой странице можно редактировать информацию о пользователе списка желаний.

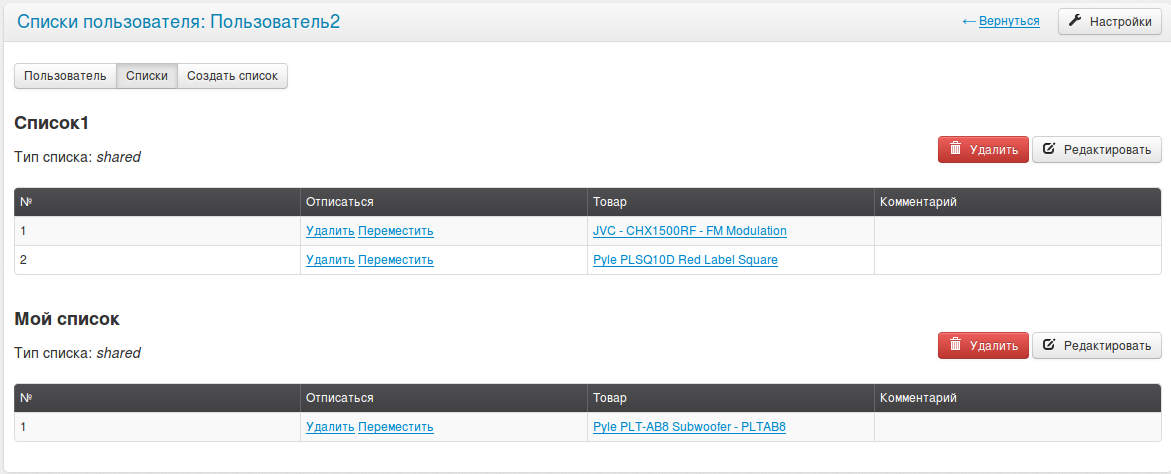


**Страница создания списка пользователя:**



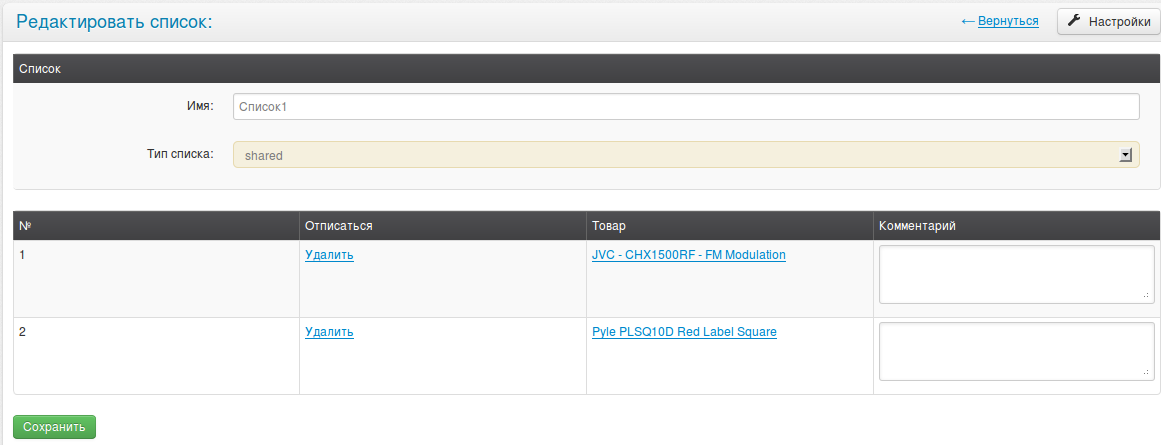
**Страница управления списками пользователя:**

На етой странице наедается возможность управлять списками пользователя: удалять, редактировать, перемещать наименования в списках и удалять их.



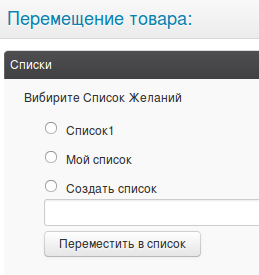
**Страница редактирования списка:**

Эта страница предназначена для редактирования одново списка желания. Здесь доступны возможности по изменению всего находящегося в списке: редактировать имя и тип списка, а также удалять наименования и изменять их комментарии.



**Страница перемещения наименования:**

Страница предназначена для перемещения наименования в другой существующий или с возможностью создать новый список желания.



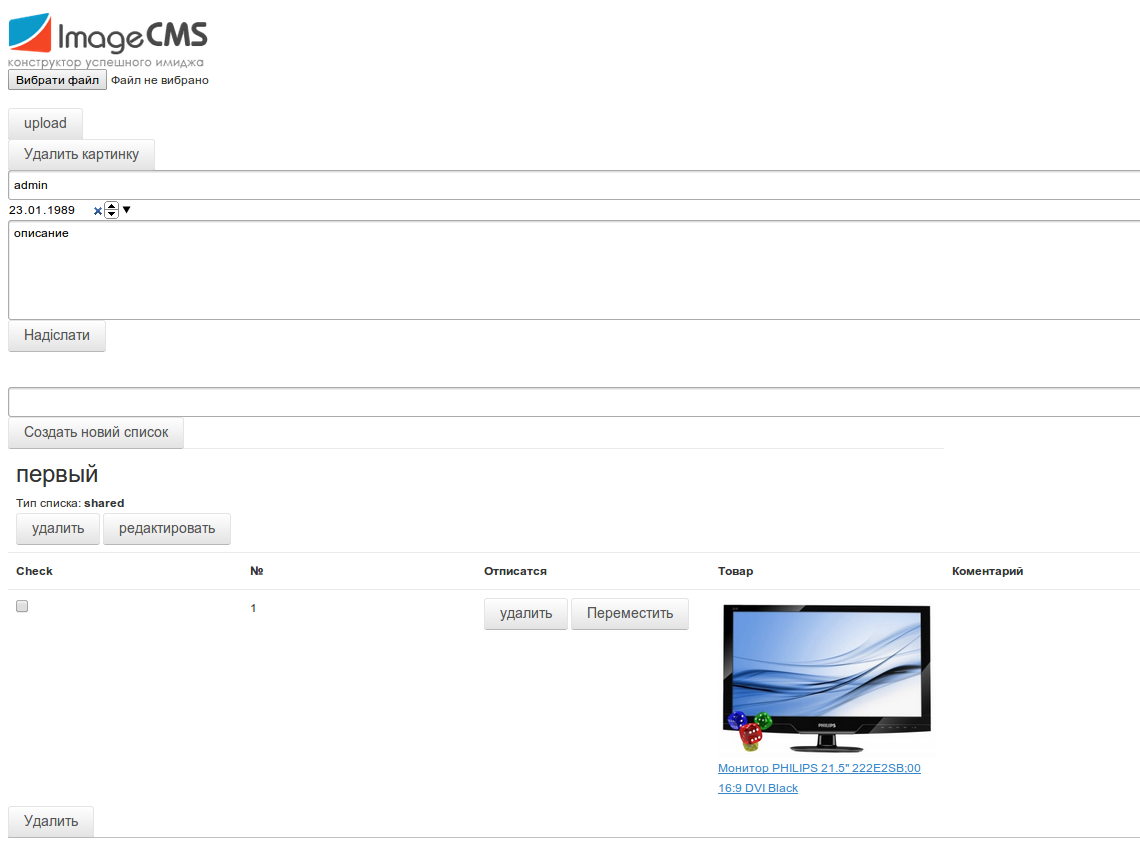
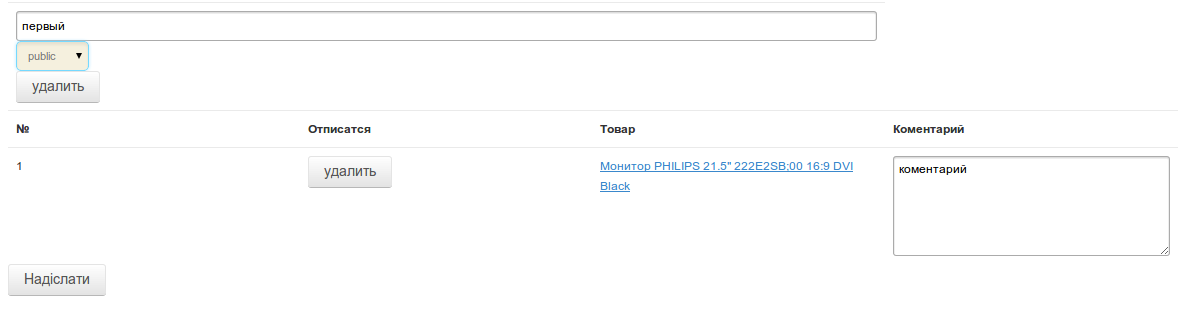
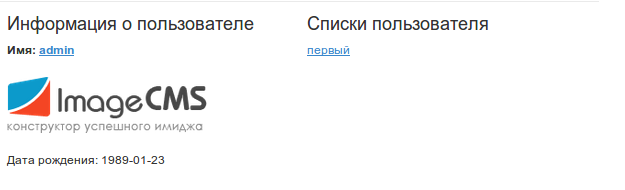
# **Site Integration**

## Вывод кнопок добавления в Список Желания

Для того чтоб пользователь мог добавить товар в Список Желания средствами модуля нужно добавить в шаблон строку:

{$CI->load->module('wishlist')->renderWLButton(***ИД\_товара***)}

## Также после установки модуля будут доступны следующие возможности:

1. Перейдя по силке site.url/wishlist пользователю будут доступен просмотр всех Списков Желания, редактирование данных, создание нового списка;
2. Нажав на кнопку “Редактировать” - вы попадаете на страницу редактирования Списка Желания;
3. На странице site.url/wishlist/all - доступен список всех *публичных* Списков Желания. Используя его пользователи смогут просмотреть что хотят приобрести другие пользователи магазина.

# **Use case**

## Use Case: #1

|  |  |
| --- | --- |
| Прецедент | Прецедент WL 1 |
| Действующие лица | Пользователь Система |
| Цель | Добавить товар в WL для не авторизованного пользователя создавая новый WL. |
| Краткое описание | Войти в систему и добавить товар создавая новый WL |

|  |  |
| --- | --- |
| Действия исполнителя | Отклик системы |
| Не авторизованный пользователь добавляет товар в WL. | Система выдает окно авторизации пользователя. |
| Пользователь авторизуется. | Система возвращается к странице просмотра или страницы товара. |
| Пользователь добавляет товар в WL. | Система дает возможность создать новый WL или добавить товар в созданный WL. |
| Пользователь создает новый WL называя его в поле для введеня названия. | Система ожидает. |
| Пользователь нажимает добавить WL. | Система создает WL с товаром в нем. |

## Use Case: #2

|  |  |
| --- | --- |
| Прецедент | Прецедент WL 2 |
| Действующие лица | Пользователь Система |
| Цель | Добавить товар у WL для не авторизованного пользователя используя существующий ВЛ. |
| Краткое описание | Войти в систему и добавить товар используя WL который уже созданный |

|  |  |
| --- | --- |
| Действия исполнителя | Отклик системы |
| Не авторизованный пользователь добавляет товар в WL. | Система выдает окно авторизации пользователя. |
| Пользователь авторизуется. | Система возвращается к странице просмотра или страницы товара. |
| Пользователь добавляет товар в WL. | Система дает возможность создать новый WL или добавить товар в созданный WL |
| Пользователь добавляет товар в существующий WL выбирая его с помощью кнопки. | Система ожидает. |

## Use Case: #3

|  |  |
| --- | --- |
| Прецедент | Прецедент WL 2 |
| Действующие лица | Пользователь Система |
| Цель | Добавить описание к существующему WL |
| Краткое описание | Описать WL который уже созданный |

|  |  |
| --- | --- |
| Действия исполнителя | Отклик системы |
| Пользователь переходит в список WL. | Система предоставляет список WL |
| Пользователь нажимает редактирования выбранного WL | Система дает доступ к избранному WL и инструменты управления им. |
| Пользователь вводит описание в поле данном для этого. | Система ожидает. |
| Пользователь нажимает сохранить изменения. | Система сохраняет изменения. |

## Use Case: #4

|  |  |
| --- | --- |
| Прецедент | Прецедент WL 4 |
| Действующие лица | Пользователь Система |
| Цель | Создать WL. |
| Краткое описание | Создать новый WL используя инструменты |

|  |  |
| --- | --- |
| Действия исполнителя | Отклик системы |
| Пользователь переходит в список WL. | Система предоставляет список WL и инструменты управления WL. |
| Пользователь нажимает кнопку «Создать WL» и заполняет поле имя. | Система ожидает. |
| Пользователь нажимает сохранить изменения. | Система создает новый WL. |

## Use Case: #5

|  |  |
| --- | --- |
| Прецедент | Прецедент WL 5 |
| Действующие лица | Пользователь Система |
| Цель | Переместить товар с одного WL к другому. |
| Краткое описание | Используя инструменты WL переместить товар. |

|  |  |
| --- | --- |
| Действия исполнителя | Отклик системы |
| Пользователь переходит в список WL. | Система предоставляет список WL и инструменты управления WL. |
| Пользователь выбирает товар в любом WL и нажимает на контекстное меню перемещения. | Система выдает список всех созданных WL. |
| Пользователь выбирает WL | Система переносит выбранный товар в выбранный WL. |

## Use Case: #6

|  |  |
| --- | --- |
| Прецедент | Прецедент WL 6 |
| Действующие лица | Пользователь Система |
| Цель | Добавить информацию о пользователе. |
| Краткое описание | Добавить информацию о пользователе которую смогут увидеть другие пользователи. |

|  |  |
| --- | --- |
| Действия исполнителя | Отклик системы |
| Пользователь переходит в список WL. | Система предоставляет список WL и инструменты управления WL. |
| Пользователь нажимает на добавлении аватару. | Система выдает дополнительное меню по загрузке аватара. |
| Пользователь выбирает файл загрузки и нажимает "Выбрать". | Система грузит аватар. |
| Система вставляет аватар в выделенной зоне для аватара. | Система ожидает. |
| Пользователь вводит свою дату рождения. | Система ожидает. |
| Пользователь вводит свое имя и фамилию. | Система ожидает. |
| Пользователь вводит описание в выделенном для этого месте. | Система ожидает. |
| Пользователь нажимает сохранить изменения. | Система сохраняет изменения введены пользователем. |

## Use Case: #7

|  |  |
| --- | --- |
| Прецендент | Прецендент WL 7 |
| Действующие лица | Пользователь Система |
| Цель | Изменение вида действующего либо не действенного WL |
| Краткое описание | Изменение вида действующего либо не действенного WL с помощью инструментов |

## 

|  |  |
| --- | --- |
| Действия исполнителя | Отклик системы |
| Пользователь переходит в список WL. | Система предоставляет список WL и инструменты управления WL. |
| Пользователь нажимает редактирования выбранного WL. | Система дает доступ к избранному WL и инструменты управления им. |
| Пользователь меняет тип с типа который указан по-стандарта на тип желающий. | Система ожидает. |
| Пользователь нажимает сохранить. | Система хранит тип WL меняя его данные. |

## 

## 

## 

## Use Case 1 - 7 в виде схемы

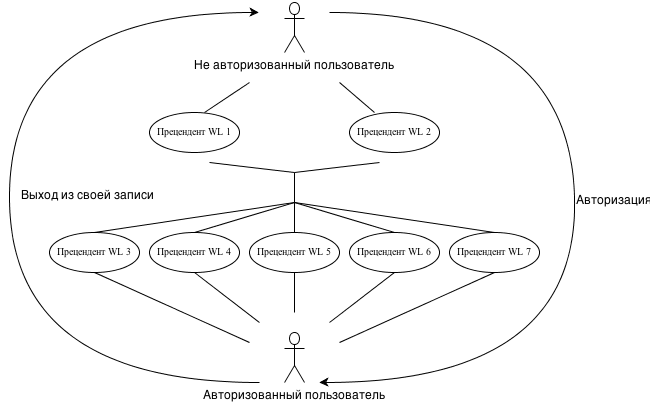


рис.1

# **Development info.**

## Класс ParentWishlist

### Properties

* **$settings** - массив, служит для хранения настроек модуля;
* **$dataModel** - массив, служит для хранения результатов методов класса;
* **$errors** - массив, служит для хранения ошибок методов класса.

### Public methods

* **all()**

(array $access = array('public'))

метод для получения всех публичных списков пользователей;

* **show()**

(string $hash, array $access = array('public'))

метод для получения публичного списка пользователя;

* **getMostViewedWishLists()**

(int $limit = 10)

метод для получения наиболее просматриваемых списков;

* **user()**

(int $user\_id, array $access = array('public'))

метод для получения публичных списков пользователя;

* **userUpdate()**

(int $userID, string $user\_name, string $user\_birthday, string $description)

метод для обновления информации о пользователе;

* **updateWL()**

(int $id, array $data, array $comments)

метод для обновления списка;

* **createWishList()**

(int $user\_id, string $listName)

метод для создания списка;

* **deleteWL()**

(int $id)

метод для удаления списка;

* **deleteAllWL()**

(int $UserID)

метод для удаления всех списков пользователя;

* **\_addItem()**

(int $varId, int $listId, string $listName, int $userId = null)

метод для добавления наименования в список;

* **moveItem()**

(int $varId, int $wish\_list\_id, int $to\_listId = '', int $to\_listName = '', int $user\_id = null)

метод для перемещения наименования;

* **deleteItem()**

(int $variant\_id, int $wish\_list\_id)

метод для удаления наименования;

* **getUserInfo()**

(int $id)

возвращает информацию про пользователя;

* **getUserWL()**

(int $userId, array $access = array('public', 'private', 'shared'))

метод служит для возвращения списка пользователя;

* **renderUserWLEdit()**

(int $wish\_list\_id, int $userID = null)

метод для возвращения списков для редактирования;

* **do\_upload()**

(int $userID = null)

метод для загрузки изображения пользователя;

* **getMostPopularItems()**

(int $limit = 10)

метод служит для возвращения наименований которых наибольшое количество в списках;

* **getUserWishListItemsCount()**

(int $user\_id)

метод возвращает количеств наименований в списках пользователя;

* **deleteItemsByIds()**

(array $ids)

метод служит для удаления наименований по их идентификатору;

* **deleteImage()**

(string $image)

метод для удаления изображения пользователя.

## API’s

## Models

* **getSettings()**

возвращает массив настроек модуля;

* **setSettings()**

(array $settings)

проводит сериализацию и сохраняет полученный массив настроек;

* **getWishLists()**

(int $userID = NULL)

получает массив Списков Желаний по ИД пользователя;

* **getAllUsers()**

получает список всех пользователей которые создавал Списки Желания;

* **createWishList()**

(int $listName, int $user\_id, string $access = 'shared')

создает Список Желания;

* **addItem()**

(int $varId, int $listId, string $listName, int $user\_id = null)

добавляет товар в Список Желания;

* **delWishListById()**

(int $id)

удаляет Список Желания;

* **deleteItem()**

(int $variant\_id, int $wish\_list\_id)

удаляет товар со Списка Желания;