Project Plan

Members

Tomas Heiskanen, 64843T, tomas.heiskanen@gmail.com
Lauri Helkkula, 62820H, lauri.helkkula@aalto.fi
Aleksi Lukkarinen, 270908, aleksi.lukkarinen@aalto.fi

Project Goal

The purpose of this project is to implement a small browser-based application allowing customers to design simple photo albums, which can then be shared with others online and/or ordered as a hard copy.

Features Planned To Be Implemented

1. A welcome page describing the application and listing some public albums.
2. Customer can log in, log out and register.
3. Customer can list all public albums and his/her private albums.
4. Customer can create new albums
5. Customer can modify existing albums which he/she owns:
    a. New pages can be added. Layout of every page can be selected from a predefined set of layouts.
    b. Existing pages can be removed.
    c. Images and captions can be changed on each page. Customer is able to upload images to the server from his/her workstation.
    d. Each of the albums can be declared as either public or private.
6. Customer can browse existing public and his/her own albums.
7. Customer can put albums to a shopping cart and finally make an order containing the selected items. The payment is made via the Simple Payments service.
8. Customer can view and modify his/her personal information (address etc.) on his/her profile page.
9. Customer can post a link to a photo album to Twitter.
10. For each album he/she owns, customer can generate a public link that can be used to link to the album when sharing online.
11. Customer can login to the service with his/her Facebook account.

Implementation Notes

- Language of the application will be English.
- The application will support only the newest Internet browsers, which have good support the necessary web standards. Supporting older non-standards-compliant browsers is an endless circle and does not fit to this small project.
    - Supported browsers are Chrome 16+, Firefox 9+, Internet Explorer 9+
- The application will require JavaScript support in the web browser.
- Customer authentication is done by Django's authentication framework by default. Additionally, customers can sign in using their Facebook accounts, in which case the actual logging in is handled by Facebook.
- Browsing of the albums will be AJAX-based, i.e. the initial loading of the album-browser web page is enough.

- Administration interface of the application will be implemented utilizing the Django Admin Site.
- The project is planned to utilize the following technologies: Python, Django, jQuery, jQuery UI, Django South and the Python Imaging Library.
- GitHub is used for version control
- The hours used are reported weekly in a worksheet
- The need for unit tests is determined individually for each feature. Django testing libraries are used (unittest, doctests)

Tentative Implementation Schedule

Deadline	Features
31.1.2012	1, 2
7.2.2012	3, 4
14.2.2012	5, 6
21.2.2012	7, 8, 9, 10, 11
28.2.2012	Last minor tweaks
4.3.2012	Deployment and the final submission

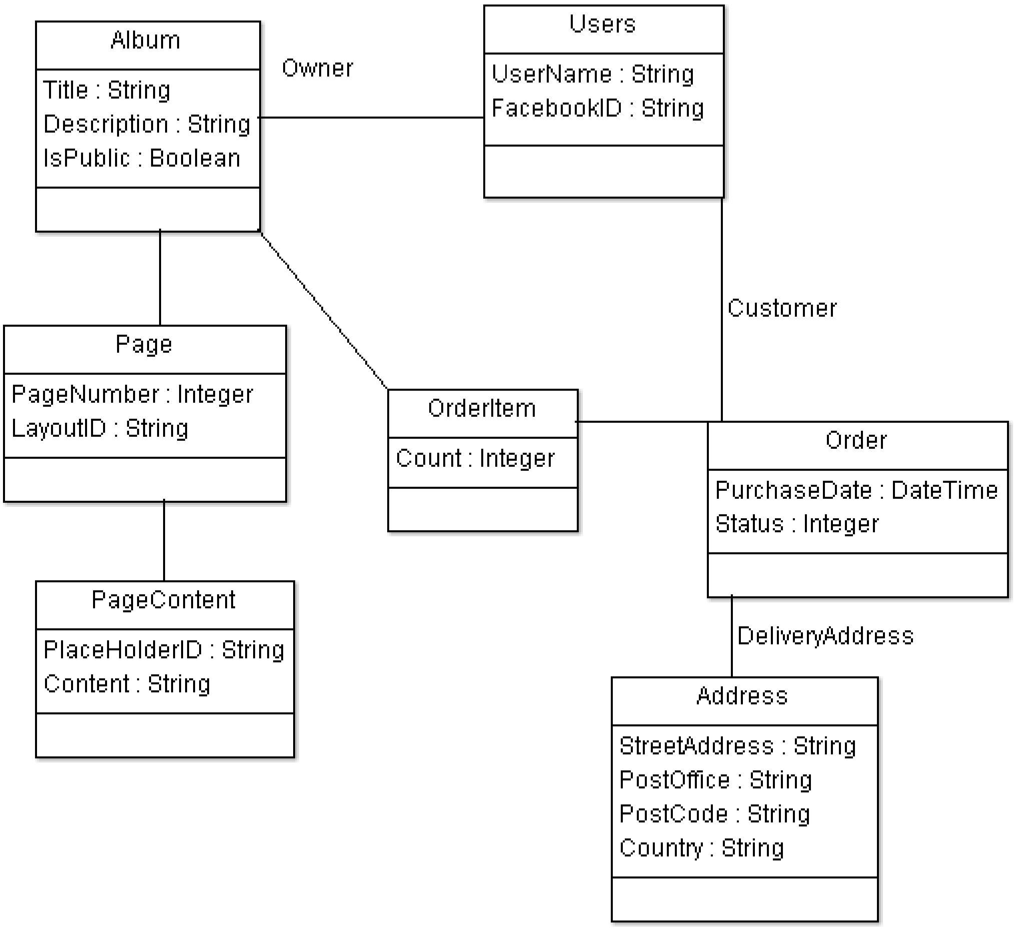
Tentative View List

welcome_page	The first view of this application, describing the application
list_all_visible_albums	Lists all albums visible to the current customer (logged in or not)
show_single_album	Allows customer to browse a single album
edit_album	Allows customer to edit a single album
log_in	Allows customer to log in with his/her username and password
log_out	Allows customer to log out
show_profile	Shows customer his/her profile page
edit_account_information	Allows customer to edit his/her personal information managed by the application
get_registration_information	Allows customer to register himself/herself into this service
edit_shopping_cart	Allows customer to edit the content of his/her shopping cart
get_ordering_information	Lets customer to enter non-product-related information required for making an order, and finally accept the order
report_order_as_succesful	Acknowledges customer about a successful order

Tentative Database Schema And Model List

The initial database schema is shown in the figure below. The columns of the tables are likely to change as features are implemented, but the tables and their relationships shouldn't change.

An album consists of multiple pages. A page has a property named LayoutID. It identifies a Django Template. The template contains some placeholders, mainly for images and text. Each placeholder is identified with an identifier. An order consists of one or more albums. Multiple copies of a single album can be ordered with one order.


Album
Title : String
Description : String
IsPublic : Boolean
Owner
Users
UserName : String
FacebookID : String
Customer
Page
PageNumber : Integer
LayoutID : String
OrderItem
Count : Integer
Order
PurchaseDate : DateTime
Status : Integer
PageContent
PlaceHolderID : String
Content : String
DeliveryAddress
Address
StreetAddress : String
PostOffice : String
PostCode : String
Country : String