

Facebook Toolbar   
for Internet Explorer

Design Specification

Lohika

2121 El Camino Real #100

San Mateo

CA 94403

**Table of Contents**

Contents

[1. Revision History 3](#_Toc224969343)

[2. Introduction 4](#_Toc224969344)

[2.1. Scope 4](#_Toc224969345)

[2.2. Related Documents 4](#_Toc224969346)

[3. Design Requirements 5](#_Toc224969347)

[3.1. Installation Requirements 5](#_Toc224969348)

[3.2. Toolbar Functional Requirements 5](#_Toc224969349)

[3.3. Non-Functional Requirements and Technical Constraints 5](#_Toc224969350)

[3.4. Documentation and Installation Requirements 6](#_Toc224969351)

[4. Design Model 7](#_Toc224969352)

[4.1. FBClientService.exe Component Design 8](#_Toc224969353)

[4.1.1. Downloading/Uploading information from/to Facebook server 8](#_Toc224969354)

[4.1.2. User information 9](#_Toc224969355)

[4.1.3. Settings 9](#_Toc224969356)

[4.1.4. Information Update Logic Implementation 9](#_Toc224969357)

[4.2. FBIEToolbar.dll Component Design 9](#_Toc224969358)

[4.2.1. Toolbar and Sidebar components Design 10](#_Toc224969359)

[4.3. Interaction Diagrams 12](#_Toc224969360)

[4.3.1. Create Toolbar 12](#_Toc224969361)

[4.3.2. Update 13](#_Toc224969362)

[5. Installation 15](#_Toc224969363)

[5.1. Installation steps 15](#_Toc224969364)

[5.2. Distribution package 15](#_Toc224969365)

[6. Development environment 16](#_Toc224969366)

# Revision History

|  |  |  |
| --- | --- | --- |
| **DATE** | **REVISION** | **DESCRIPTION OF CHANGES** |
| 23 Jan 2009 | 0.1 | First Draft |
| 30 Jan 2009 | 1 |  |
| 11 Mar 2009 | 1.1 | Updated according to pilot phase design |
| 16 Mar 2009 | 1.2 | Final update |

# Introduction

## Scope

This document presents a detailed solution to the problem defined by the Facebook IE Toolbar Software Requirements Specification.

## Related Documents

1. Facebook IE Toolbar SRS.docx
2. <http://wiki.developers.facebook.com/index.php/API>

# Design Requirements

This section contains summary of requirements described in [1].

## Installation Requirements

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Description** |
| INST.1 | Installation from Windows installation | InnoSetup installation used to install toolbar and register necessary components under Win XP and Vista |
|  |  |  |

## Toolbar Functional Requirements

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Description** |
| TLB.1 | Login/Logout | User should be able to login/logout to Facebook from toolbar. Appropriate html dialog will appear for login, which will request username and password and after that session would be established. |
| TLB.1.1 | Save Session | Application should store session to stay logged to the Facebook site permanently. |
| TLB.2 | Quick Links | There is a number of quick links available on the Toolbar: Profile, Friends, Networks, Inbox, Photos, Notes, Groups, Events, Posted Items, Account, Privacy, Home, Go to Profile. They should redirect browser to corresponding Facebook page. |
| TLB.3 | Set Status | Change user status. |
| TLB.4 | Get notified by Icons | Show notifications about new messages, invitations, etc on the toolbar |
| TLB.5 | Get notified by popup window | Show popup notifications about new messages, invitations, friends status changes, etc |
| TLB.6 | Share Content | Show popup which propose to share content of the current page to friend |
| TLB.7 | Search Facebook | Show popup which shows friends whose names match to text typed by user in search input box |
| SDB.1 | Facebook friends sidebar | Panel on the left side of browser which contain list of friends with picture, status message, status update time and links to actions like poke friend, write on the wall, etc |
| SDB.1.1 | Sorting | Ability to sort friends on sidebar. Sort options: by name, status update time and update time |
| SDB.1.2 | Toolbar Settings | Dialog which allow to modify toolbar notification settings |

## Non-Functional Requirements and Technical Constraints

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Description** |
| NFR.1 | Look and Feel | Toolbar should be similar in UI to Firefox toolbar. |
| NFR.2 | Communication Interface to Facebook | Toolbar should use the same interface (HTTP) as original Firefox toolbar to interact with the Facebook system. |
| NFR.3 | Security Requirements | All personal information of the user should be protected to not allow other persons read it. |
| NFR.4 | Single communication component | All instances of IE Toolbar should communicate to Facebook system using single communication component, which handles session and decrease internet traffic |
| NFR.5 | System configuration supported | MS Windows XP/Vista  MS IE 6.0 and 7.0 |
| NFR.6 | Development language/environment | C++, MS VS 2008 |
| NFR.7 | Code conventions | All code should follow Facebook C++ Coding Conventions and C++ Style. |

## Documentation and Installation Requirements

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Description** |
| DIR.1 | Installation from Windows installation | User has ability to install toolbar under Windows XP and Vista easily using installation program |
| DIR.2 | Documentation | The final product should be documented both inline as well as externally in the same manner the current Firefox toolbar is. |
| DIR.3 | Product source code | Product sources should meet Facebook code conventions and coding style for C++ language |

# Design Model

High Level Architecture is shown on the figure below.



Toolbar and Sidebar are implemented as COM components with interfaces defined by Microsoft for IE Toolbars. Both implemented inside COM Inproc Sever FBIEToolbar.dll. Toolbar and Sidebar components obtain and update all Facebook information (user details and status, his friends’ details) using FBClientService.exe. FBClientService.exe is a COM Outproc Local server, which implements all logic for communication and information retrieval/uploading from/to Facebook server. Such architecture allows keeping all Toolbars synchronized and updated simultaneously, and minimizes network traffic between client and server. Also FBClientService.exe uses FBClientServiceUI.dll for UI purposed: display login dialog and popup notifications.

## FBClientService.exe Component Design



As stated above, FBClientService.exe is a COM Local Outproc Server, which implements COM component with IFBClientService interface. It sends notifications about data updates to its clients using dataUpdated event in IFBClientServiceEvents interface.

The component implements all logic of communication with Facebook server, update on timers and session storage.

### Downloading/Uploading information from/to Facebook server

Logic of communication and information retrieval/uploading from/to Facebook server is implemented in CommunicationService class, which uses RequestsComposer to build requests to server and ResponseParser to parse response. Communication is performed in separate thread wrapped by CommunicationLoop class. UpdatesComparator used to compare current and new data to build notifications.

Communication classes implement following functionality:

1. Login to Facebook server
2. Calculate session key and session secret using MD5 algorithm to use them for following operations
3. Downloads from Facebook server information about logged-in user and number of events (pokes, messages etc) and fills it to instances of ServiceData and UserData classes
4. Downloads list of user friends, their attributes and fills instances of UserData class with this information
5. Downloads friend’s album data.

### User information

All information about current user, his status, number of different events etc is kept in instances of ServiceData and UserData classes. All information about user’s friends is kept as a collection of UserData objects as shown on the figure below.

UserData has same attributes as fbIFacebookUser interface, which is used in original Facebook Toolbar for FireFox..

### Settings

ToolbarSettings class is responsible for keeping and storing/loading all toolbar settings. SessionInfo wraps user session data.

Toolbar settings and user session are stored into the Windows registry.

### Information Update Logic Implementation

Facebook information update is made every 30 seconds minutes by timer. Communication with Facebook server is made by CommunicationLoop class in a separate thread. CommunicationService uses HTTP as a communication protocol. Detailed information about the protocol and request/response formats can be found in [2]. All logic regarding composing requests and parsing response is implemented in RequestsComposer and ResponseParser classes.

CommunicationService fills new instances of ServiceData and UserData classes. Then these new objects are compared with old ones and based on difference using UpdatesComparator class, ClientService class object sends corresponding dataUpdated events, one per each type of different information.

## FBIEToolbar.dll Component Design

As stated above, FBIEToolbar.dll is a COM Inproc server, which implements Toolbar and Sidebar COM components. Both Toolbar and Sidebar components implement standard MS IE interfaces (IDeskBand, IObjectWithSite, IInputObject, IPersistStream) to interact with MS IE.

When created, Toolbar and Sidebar uses UserDataObserver instance to obtain instance of FBClientService component, obtain from it all information presented to user and subscribe to FBClientService update events. By these events, they receive notifications when any information has been updated and update their UI views accordingly. See several samples of interactions in section 4.3.

Because any COM interaction is time consuming, to minimize it we create single event sink point, implemented in ClientServiceEventsSink class. Single instance of this class subscribes to FBClientServices update events and then redistribute these events to Toolbar and Sidebar using Windows messaging architecture.

In the same time, Toolbar and Sidebar obtain all information they require from FBClientService using UserDataObserver class, which covers communication into separate thread and does not affects UI.

RuntimeContext class is used to access toolbar and sidebar instances created for current IE instance. 

### Toolbar and Sidebar components Design

**Toolbar and Sidebar**

Toolbar and Sidebar general structure is shown on the figure below. They use standard and customized MFC controls to display their content.

**Share Content** **Window**

Share Content Window and its functionality are implemented as standard MS IE popup dialog window, redirected to corresponding Facebook page.

**Toolbar Settings Dialog**

Settings Dialog is implemented as a standard modal MFC dialog with corresponding controls. It uses ToolbarSettings class for settings information retrieval and update.

**Toolbar window control**

To implement FireFox like look&feel toolbar, custom toolbar control is implemented by ToolbarWindow class. This supports custom controls, buttons with different width and position.

**Search box**

ToolbarEdit class provides custom draw functionality, used by descendant FriendsSearchEdit control to implement FireFox like search box. Also FriendsSearchEdit implements show/hide search popup functionality, handle focus to be shown even when popup is visible, etc/

**User Status box**

This control, implemented by UserStatusEdit class, sets status functionality and display “set status…” text when user status is not set.

**Html views**

There are several html views used in the product:

1. Sidebar html view
2. Search popup html view
3. Notifier
4. Login dialog

Notifier and Login dialog uses GenericHtmlView class which extends MFC CHtmlView class with custom functionality to handle content loading and configuring UI behavior (disable context popup menu, etc).

The most complex UI control is displaying friends list with names, pictures, statuses and links as in original FireFox toolbar. This control used by sidebar and search popup with some customizations. Such functionality implemented in BaseHtmlListView control, which render html code to display friends, uses javascript to perform keyboard navigation, process predefined variables in html content to load CSS from resources, etc. Html code for friends list is generated based on XML/XSLT transformation, CSS styles are stored in resources - this provides ability to customize friends list without changing C++ code.

SidebarHtmlView extends base functionality with custom actions handling (open web pages in current web browser).

SearchPopupHtmlView handles the following situations:

1. one friends is found - shows different view);
2. search string is empty – shows “See all friends” view, and opens sidebar when clicked on inside view
3. no friends found – propose to search for friend on web site



## Interaction Diagrams

Following diagrams show most complicated types of interactions between FBClientService.exe and FBIEToolbar.dll components.

### Create Toolbar

The diagram below shows the process of Toolbar creation, when new instance of IE is starting.



### Update

The diagram below shows method calls sequence executed, when Facebook user data is updated by timer. First diagram shows interactions going inside FBClientService.exe component and second one show interactions in FBIEToolbar.dll and FBClientService.exe, but with most focus on Toolbar logic.





# Installation

Installation is implemented based on InnoSetup installation solution. Setup configuration is covered by single settings file: setup.iss. See comments in this file for details.

## Installation steps

Installation includes the following steps:

* Copy product files to destination folder selected by user
* Register all COM components which require registration
* Configure Internet Explorer to show IE Toolbar on next start
* Ask to restart computer: mandatory for Vista, for XP sometimes IE does not recognize new sidebar components without computer restart

## Distribution package

Installation contain the following files:

|  |  |
| --- | --- |
| File name | Description |
| FBClientService.exe | Communication component |
| FBClientServiceUI.dll | Communication component UI helper library |
| FBIEToolbar.dll | Toolbar and Sidebar components |

# Development environment

You have to perform next steps to configure Facebook toolbar for IE build environment:

1. Install Visual Studio 2008
2. Install Visual Studio 2008 Feature Pack.   
    You can download feature pack using this url:  
   <http://www.microsoft.com/DownLoads/details.aspx?FamilyID=d466226b-8dab-445f-a7b4-448b326c48e7&displaylang=en>
3. Install boost (version 1.35.0).  
   You can download using this url:  
   <http://sourceforge.net/project/showfiles.php?group_id=7586&package_id=8041>
4. Install log4cplus (version 1.0.3 rc9)

You can download using this url:

http://downloads.sourceforge.net/log4cplus/log4cplus-1.0.3-rc9.tar.bz2?use\_mirror=heanet

1. Configure Visual Studio 2008 with boost folder:
2. Open Visual Studio 2008.
3. Select menu item Tools->Options.
4. Navigate to the “Projects and Solutions” -> “VC++ Directories” in the left hand treeview.
5. Select the “Include files” in the “Show directories for” combobox.
6. In the include folders textbox press insert new and add path to the boost root folder.
7. Press OK.
8. Configure Visual Studio 2008 with log4cplus folder:
9. Open Visual Studio 2008.
10. Select menu item Tools->Options.
11. Navigate to the “Projects and Solutions” -> “VC++ Directories” in the left hand treeview.
12. Select the “Include files” in the “Show directories for” combobox.
13. In the include folders textbox press insert new and add path to the log4cplus root\include folder.
14. Press OK.

1. Install ant (version 1.7).  
   You can download using this url:  
    <http://ant.apache.org/>
2. Unpack toolbar sources
3. Modify build.cmd:
   1. set path to ant (see comments in build.cmd )
   2. set path to VS2008 (see comments in build.cmd )