



Project no. 034595

SELF  
*Science, Education and Learning in Freedom*

Instrument: SSA - Specific Support Action  
Thematic Priority: IST-2005-2.5.5 – Software and Services

**Deliverable Report D6**  
**SELF Platform Development:**  
**localised and tested system**

Period covered:  
from July 1<sup>st</sup>, 2006 to July 31<sup>st</sup>, 2008

Due date: April 30<sup>th</sup>, 2008  
Date of first submission: April 29<sup>th</sup>, 2008  
Revision date: September 18<sup>th</sup>, 2008

Start date of project:  
July 1<sup>st</sup>, 2006

Duration:  
2 years

Organisation name of lead contractor for this IR  
Homi Bhabha Center for Science Education  
Dr. Nagarjuna G.

Revision version 1.0

<b>Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)</b>		
<b>Dissemination Level</b>		
<b>PU</b>	Public	PU
<b>PP</b>	Restricted to other programme participants (including the Commission Services)	
<b>RE</b>	Restricted to a group specified by the consortium (including the Commission Services)	
<b>CO</b>	Confidential, only for members of the consortium (including the Commission Services)	

# Table of Contents

Acknowledgements .....	3
1 Introduction .....	4
2 Roadmap and Development Methodology .....	5
3 Setting up the Server Infrastructure .....	6
3.1 Technical specifications .....	6
3.2 Setup and maintenance .....	6
3.3 Management and maintenance .....	7
4 Implementation .....	8
4.1 Platform Implementation .....	8
4.2 Human Interface Design .....	9
4.3 List of Main Features Implemented .....	10
4.4 Implementation of Features for Phase II .....	28
4.5 Features for Phase III .....	28
5 Developers and Users Guide .....	30

## Acknowledgements

This document has been collaboratively written by the SELF Project team. It is a summary of the implementation of Deliverables D4 and D5<sup>1</sup>.

The Team has held many discussions on various alternatives of shaping the Platform, and implementation details.

The team mainly consisted of:

Developers team: Prasanta Barua, Bipin Apandkar, Debarshi Ray, Rakesh Pandit, Rajiv, Krishnakant, Divya, Meena Kharatmal and Nagarjuna G..

Developers team leader: Nagarjuna G.

Code review by: Matías Bordese

Requirements team: Federico Heinz, Meena Kharatmal, Dragoslava Pefeva, David Jacovkis, Nagarjuna G., David Megias and Wouter Tebbens.

All participants are gratefully thanked for their valuable inputs and for their active participation.

---

<sup>1</sup> <http://selfproject.eu/en/project/results>

# 1 Introduction

This document is a brief report of the development of the SELF Platform, as specified in the work package WP6.

The main task is to develop a graphical representation of the data and dynamics of the SELF system, based on the technical specifications recommended by WP 5, and the data model based on WP 3, to set up the Technology Expert Group for discussing the development process, to test the features implementation.

The graphical representation of the model of the SELF Platform was submitted to EC as Deliverable D5 “SELF Platform Development: System design and representation and beta release” at the end of May 2007. This report mainly consists of the description of the results of WP6, and the on-going work.

The development of SELF Platform is guided and as well as constrained by the following principles:

- a) It shall be a web-based platform for the collaborative authoring, remixing and organising of educational materials in conformance with existing open standards like SCORM.
- b) It shall be a teacher (author) centric platform facilitating the building of learner centric course material.
- c) It shall comply with the SELF Founding principles<sup>2</sup>

---

2 The SELF Founding Principles are defined here: <http://selfproject.eu/en/project/principles>

## 2 Roadmap and Development Methodology

The implementation of the Platform was executed collaboratively by involving all the partners, though the fabrication of the system and software development was mainly done by the knowledge.org lab of the Indian partner in the consortium, Homi Bhabha Centre for Science Education, part of Tata Institute of Fundamental Research, in Mumbai.

The actual development of the platform began after February 2007, after the recruitment of full time software engineers specially dedicated to the execution and development of the Platform. A broad roadmap was made keeping in mind the first beta release of the Platform scheduled for July 2007.

### Formation of the Technical Expert Group (TEG)

The process of converting the high level specifications of the Platform required constant deliberations with all members of the team. The team is geographically spread all over the world, into three main continents, Argentina, Europe and India, except for the main software developers who are all stationed in Mumbai, India. Due to the collaborative nature of the project, one of the main means of communication for specifying the use cases, reworking the requirements of the Platform, technical choices and so on, it was felt necessary to create a Technology Expert Group (TEG), which communicates mainly over a mailing list.

Most of the technical deliberations, testing results were discussed in this forum. Considering that the Platform development adopted an agile software development methodology, with more number of frequent releases, testing, reframing of the requirements and specifications by the TEG, and re-implementation, the forum turned out to be a very important medium. Main issues regarding the implementation were discussed and resolved collaboratively in this space.

### Project Management and Issue Tracking System

As the SELF Platform is an internationally spread collaborative project and also a free software project, which made it necessary to set up a community based project management for the code development. The SELF Platform was registered as a non-GNU project at Savannah<sup>3</sup>, a community portal offering all the facilities for project management, bug tracking, download, issue tracking, documentation, concurrent version control system for publishing snapshots of the software and documentation, mailing list management, and also to coordinate among the developers.

An open mailing list for the developers of the Platform was created at Savannah, keeping in mind the possibility of inviting the entire community of free software developers to take part in the development, use and testing. This will remain a permanent place where the development will continue even after the EC-funded period of the SELF project ends in July 2008.

---

<sup>3</sup> <http://savannah.nongnu.org/projects/self-platform/>

## 3 Setting up the Server Infrastructure

The server infrastructure of the SELF Project consists currently of two main servers, *bselfish* and *cselfish*. *bselfish* is the production server that can be reached in <http://selfplatform.eu>, while *cselfish* is used for testing the latest code and can be reached in <http://testing.selfplatform.eu>.

*bselfish* is sponsored by Sun Microsystems<sup>4</sup> and AMD, and its collocation is sponsored by XS4ALL<sup>5</sup>. *cselfish* is sponsored by AT Computing<sup>6</sup>, and is hosted in AT's datacenter.

### 3.1 Technical specifications

These are the basic technical specifications of the two servers:

#### *bselfish.selfplatform.eu* - 194.109.117.210

- Sun Sun Fire X2100 M2
- Processor: Dual-Core AMD Opteron 1214
- HDD: 2x 250GB SATA II
- RAM: 6GB DDR2 SDRAM
- Collocation: XS4ALL datacenter, Amsterdam

#### *cselfish.selfplatform.eu* - 80.69.84.225

- Dell PowerEdge 2950
- Processor: Quad Core Intel Xeon E5335
- HDD: 2x160GB SATA II
- RAM: 4GB DDR2 SDRAM
- Collocation: AT Computing datacenter, Amsterdam

### 3.2 Setup and maintenance

Both SELF servers run exclusively Free Software applications, complying with the philosophy and founding principles of the SELF Consortium.

#### 3.2.1 Operating system and basic software

The operating system in both the machines is Debian GNU/Linux 4.0, the latest stable version.

Storage is configured in software RAID1, also known as mirror mode. This configuration provides tolerance to logical and physical errors in the disks, reducing the risk of downtime and data loss. Read performance is good, but write performance can be affected by the fact that all data is written twice.

In order to improve the performance and security of the servers, the necessary applications were added in an individual basis after a basic installation. Besides the basic networking services, the following applications are required for the SELF Platform:

- Python 2.4
- Zope + Plone

---

<sup>4</sup> <http://nl.sun.com/>

<sup>5</sup> <http://xs4all.nl>

<sup>6</sup> <http://atcomputing.nl>

- Postgresql

### 3.2.1 Virtualisation

The *cselfish* server runs *Xen*<sup>7</sup>, a virtualisation engine with which several virtual machines in the same physical server with a small performance overhead.

Virtualisation offers many advantages for a testing server. It can provide different testing environments in a single physical machine, and it allows for a fast recovery from broken configurations. In the case of the SELF Platform, it will allow testing the distributed database system using a single server with several virtual machines connected by a virtual network.

### 3.3 Management and maintenance

Management of both servers is done remotely via secure shell connection (SSH). Both servers provide remote consoles as well, providing access to the system in case of a loss of network connectivity.

Maintenance of the servers basically consists of software updates, usually security patches. The availability of updates, particularly related to security, in Debian's official repositories is automatically checked on a daily basis.

---

<sup>7</sup> <http://xen.org>

## 4 Implementation

Based on the UI design, and the technical specifications several features of the platform are implemented, though several other features are still to be completed. The Platform is live at <http://beta.selfplatform.eu>, and the following list of features and the screen shots are taken from the live site in this report.

### 4.1 Platform Implementation

The development of the platform, which started in February 2007, underwent three iterations.

#### First Iteration

During the first implementation the focus was mostly on getting the data model right. Though the initial development took off by following the schema developed by the Learning Standards Expert Group<sup>8</sup> (SCORM model derived from aTutor), several additional schematic changes were made to ensure that the Platform supports the SCORM 2004 3<sup>rd</sup> edition specification. A fairly detailed account of the data model is presented in the deliverable D5.

After preparing the schema for the knowledge base, pilot views were made to build the entire course and course organizer. This implementation served the purpose of communicating to the HI designer of the problems to be solved. The other purpose it served was to inform ourselves that we are on the right track with the implementation.

Two versions of course organizers were made, one using a drag and drop feature implemented using Javascript, and another using simple selection with action based buttons. The former was dropped later, since this does not provide accessibility.

During this iteration, a translation interface, as well as an atomizer were tried. This initial exercise also helped in understanding the requirements more clearly, and after obtaining the feedback from the SELF-test team, the Learning Materials team, and obtaining the first drawings of the Human Interface Design (HID), we began the second iteration.

#### Second Iteration

The views were reimplemented based on the HID during this phase. Additional changes in the data model to incorporate shelf manager and file manager (resource pool) were made. Another substantial change to the knowledge base schema was to create provisions for advanced metadata, such as sequencing and adaptive rollback.

The results of this iteration were released when the Platform was launched on July 2007.

During this phase one of the central requirements for the Platform, namely, version control was implemented. At this stage we realized serious problems in performance, and scalability, indicating that the storage of knowledge base needed rethinking.

After a thorough analysis by the developer team, it was realized that the complex relations and increasingly large number of snapshots that are getting stored in the Zope object data base (ZODB) cannot scale well unless the storage of the objects is shifted either to the file system or to an external database.

The knowledge base is based on a distributed networking database, GNOWYS; the version of GNOWSYS that was used till the second iteration was depending on the native storage supported

<sup>8</sup> <http://selfproject.eu/en/LSEG>



by Zope. Rectifying this problem was already in the roadmap of GNOWSYS. Having encountered the problem mentioned above, the GNOWSYS developers began to solve the scalability and performance problem as a priority. This process started in November 2007 and was completed in December 2007. The new backend uses PostgreSQL for storage, with additional features to create snapshots of every change made to the knowledge base both in form as well as in content in a non-redundant way. Taking this as an opportunity, the API of the knowledge base was simplified so that development can become easier.

### **Third Iteration**

The third iteration of the Platform building process started after migrating the storage of the knowledge base in the new backend made available after December 2007. The nature of the work mainly consisted in making new skins (templates) that connect the presentation of the Platform with the migrated knowledge base.

There will be, we are sure, more iterations, as we go on. In the following section the results of the third iteration are presented. Since several of the basic features can be accessed through a web interface, we present the results in a visual medium including screen shots taken from the live site. The result of the third iteration is now accessible at <http://beta.selfplatform.eu> and the uptodate version of the sourcecode of the Platform can be viewed and downloaded from CVS:

<http://cvs.savannah.nongnu.org/viewvc/?root=self-platform> or

```
cvs -d:pserver:anonymous@cvs.sv.gnu.org:/sources/self-platform co
```

### **4.2 Human Interface Design**

The HI design was subcontracted to an independent designer Mr. Niyam Bhushan, who uses free software for the art work. Considering that the elearning applications' internal knowledge base is very complex, masking such complexity is one of the major challenges of defining the UI design. Each learning object, can have about 106 different kinds of attributes, and about 12 different kinds of relations. And whenever such a learning object is modified, the knowledge base keeps all the snapshots in a non-redundant way. Since several users can modify the course collaboratively, and the users are not expected to know the complexities of the technical standard (SCORM) that SELF Platform implements, one of the main challenges was not to intimidate the user with the complexity and present a simple interface without loosing the main features. Minimalism is employed in the design, so that the rich features of the platform do not overwhelm the user.

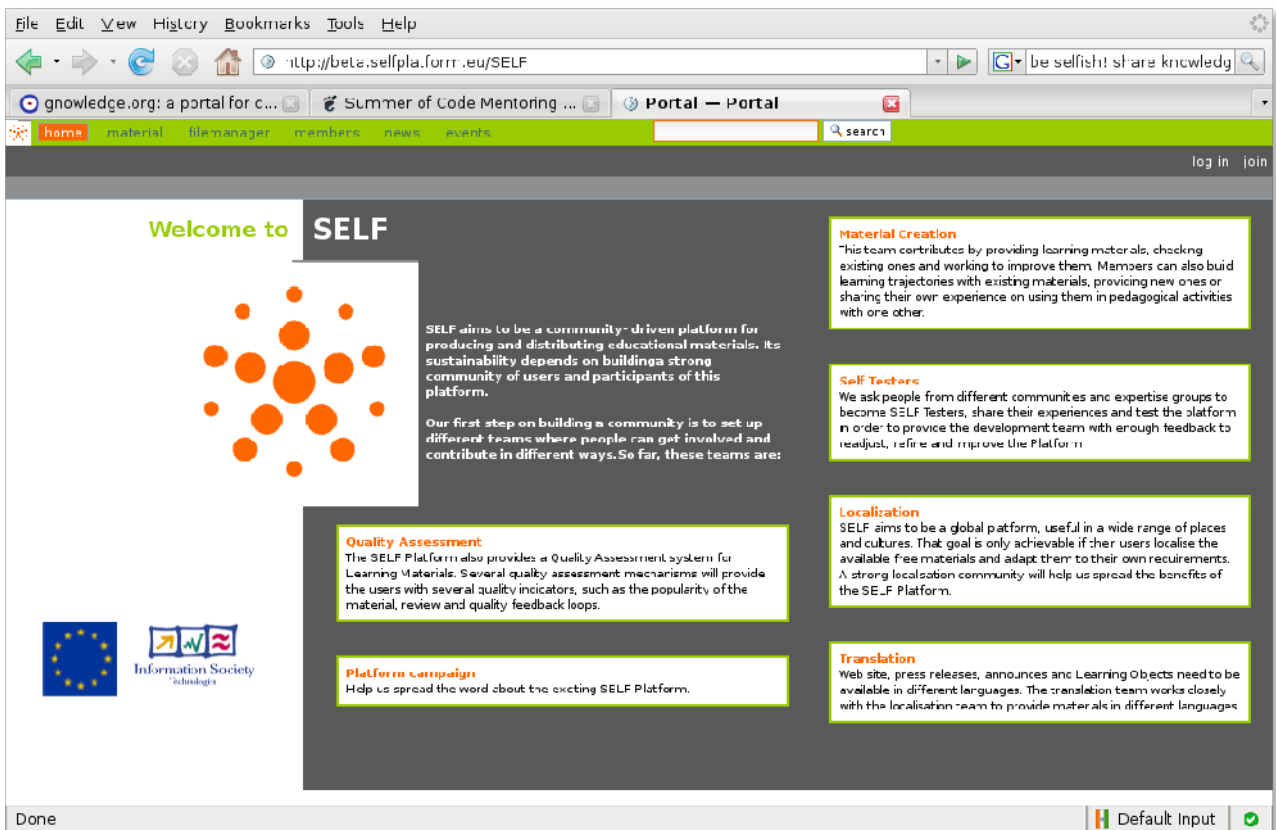
All the drawings with sources were uploaded at [www.gnowledge.org/niyam](http://www.gnowledge.org/niyam).

### **4.3 List of Main Features Implemented in Phase I**

- Backend
  - Building the Knowledge Base
  - Inbuilt Version Control of learning objects and organization
- Collection Manager
  - Adding and Editing Collections
  - Deleting Collections
  - Adding Lessons
- Course Manager
  - Adding and Updating Lessons
  - Viewing Versions
  - Course Components
    - FAQ
    - Glossary
    - Tests
    - Links and References
  - Course Organizer
  - Image Composition
  - Adding to Shelf
- File Manager
  - Uploading files
  - Downloading files
  - Searching Resources
  - Adding to Shelf
  - Importing SCORM Files
  - Information View
- Shelf Manager
  - Recombination of Lessons
  - Searching resources
  - Adding and deleting items
  - Information View
- Version Manager
  - Version Tree View
  - Preview selected versions
  - Compare selected versions
  - Graphical view of changes, insertions and deletions
- Basic Search
  - Searching based on Title and Description of learning objects and resources
  - Information View
  - Add to Shelf
  - Calculating Links to Course, Collection and Shelf Managers

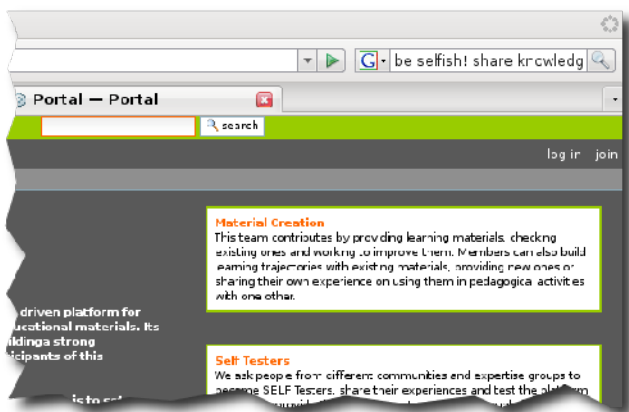
Some of the main Screenshots are given below. Platform is live at <http://beta.selfplatform.eu>

## Opening page



The current look of the frontpage of SELF Platform is meant to inform the user about the project; eventually the opening page will contain information about the activity at the site, informing the featured courses, highly rated users list, recently added lessons, recently edited lessons, news, messages etc.

## Join Page



The opening page on the right top corner contains links to 'join', inviting a user to be part of the platform, and a 'log in' link for those who are already registered users.

## Registration

File Edit View History Bookmarks Tools Help

http://beta.selfplatform.eu/SELF/j

knowledge.org: a port... Summer of Code Ment... Portal — Portal

accueil material filemanager members news events rechercher

s'identifier s'inscrire

accueil

### Formulaire d'inscription

Informations personnelles

**Nom complet**  
Saisissez votre nom complet, par exemple Jean Dupont.

**Identifiant** ■  
Indiquez l'identifiant que vous souhaitez utiliser, généralement quelque chose comme « jdupont ». Il ne doit contenir ni espace ni caractère spécial (en particulier, il ne doit pas contenir de caractère accentué). Les noms d'utilisateurs et les mots de passe respectent la casse. C'est ce nom qui vous servira à vous identifier sur ce site.

**Adresse électronique** ■  
Indiquez une adresse électronique. Cela vous permettra de recevoir un rappel de votre mot de passe en cas d'oubli. Nous respectons votre vie privée et ne communiquerons pas cette adresse à des tiers.

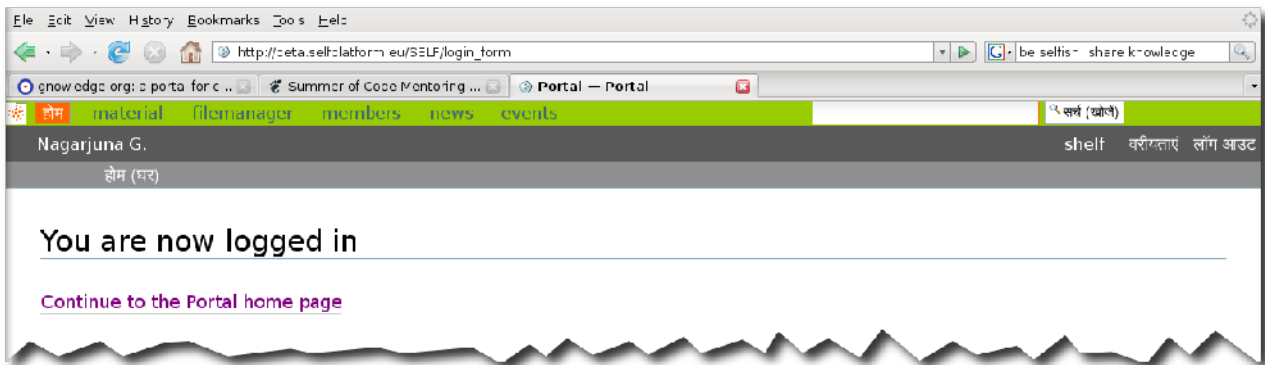
**Mot de passe** ■  
Saisissez le mot de passe de votre choix (5 caractères minimum).

**Confirmer le mot de passe** ■  
Saisissez à nouveau le mot de passe pour confirmation.

Done Default Input

*Registration form for new users. This form is available in many languages. Based on the default language set in the user's browser most users are welcomed in their browsers' language. Many other platform specific forms are not yet internationalised, but will be eventually.*

## Logged in message!



After logging in, the name of the user is displayed below the menu bar on the top left corner. Already translated messages will automatically appear in the default language of the user.

## Change Password



Logged in users can change their password through the 'Preferences' hyperlink on the personal pane.

## Forgotten Password

material filemanager members news events zoeken

inloggen lid worden

home

### Wachtwoord vergeten

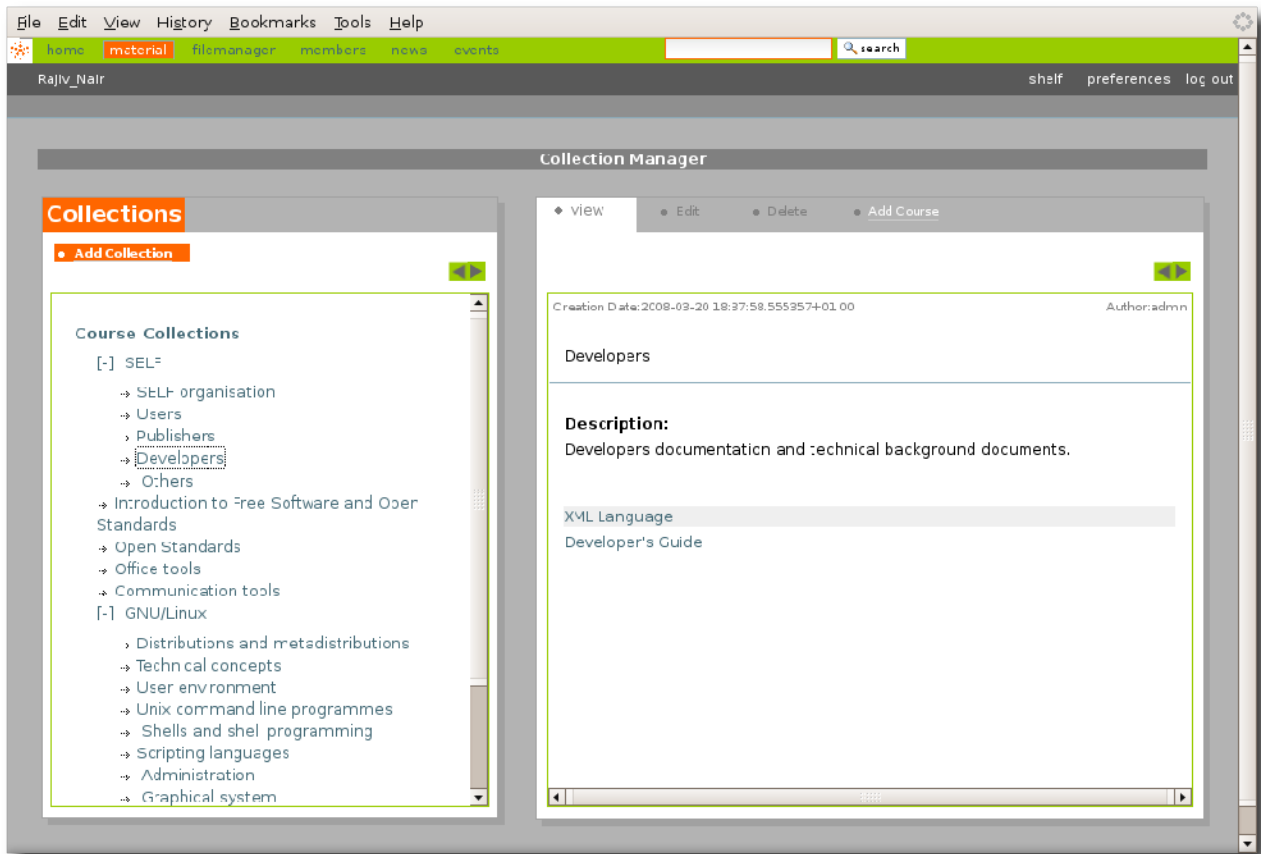
Om veiligheidsredenen kan het wachtwoord niet per e-mail naar u worden verzonden. U kunt uw wachtwoord wijzigen door het onderstaande formulier in te vullen. Wij sturen een e-mailbericht naar het e-mailadres dat u tijdens de registratie heeft opgegeven. Dit bericht bevat een link naar een formulier waar u het wachtwoord kunt wijzigen. Indien u bijvoorbeeld uw gebruikersnaam niet meer weet of uw e-mailadres is gewijzigd, stuur dan een e-mail naar [Portal Administrator](#).

**Mijn gebruikersnaam is**

*The Platform offers the feature to reset the password when a user forgets.*

*The user can also use the user preferences form to choose the preferred language.*

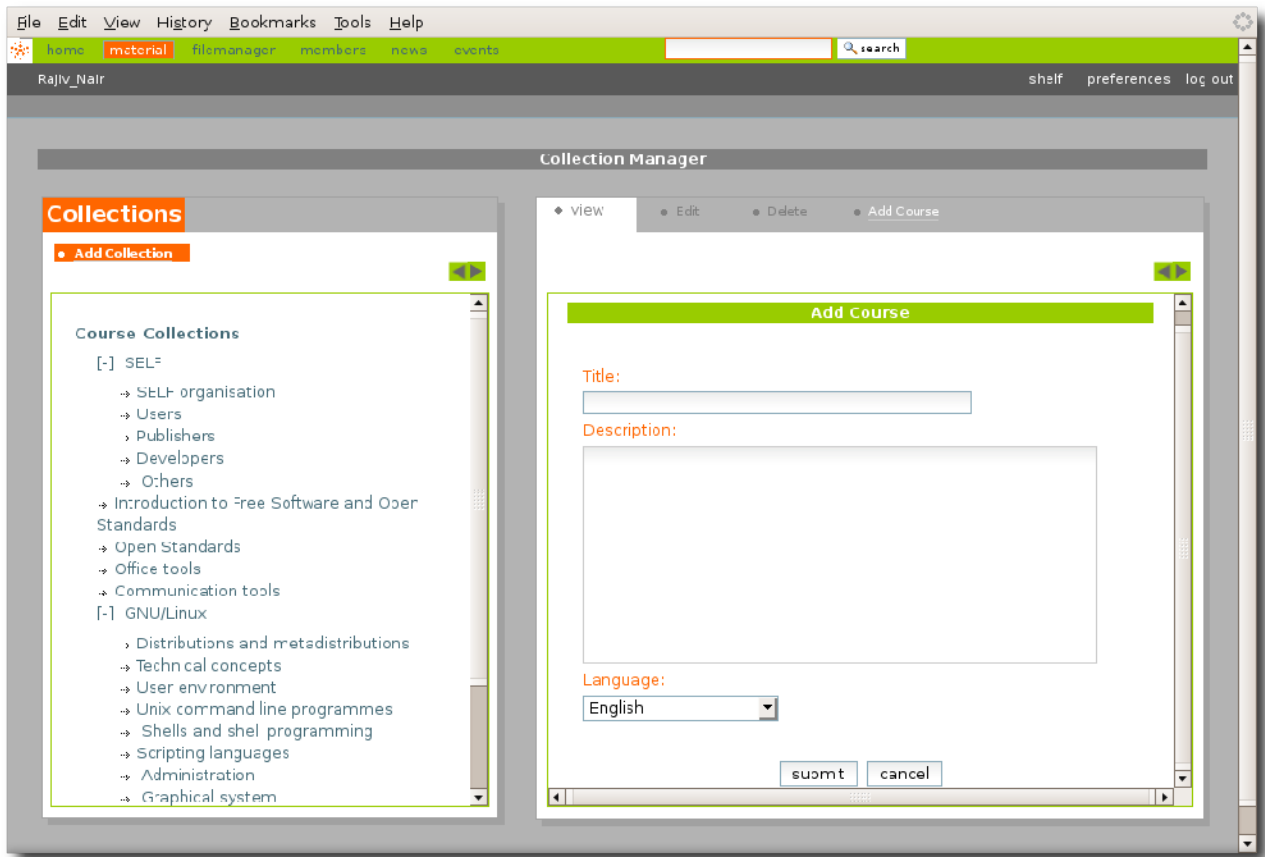
## Building the Knowledge Base Begins with Adding Collections



*The initial step of building knowledge base starts with defining the collections under which the courses can be classified. The collection manager can also be used as the site map, most of the resources are organized through the collections*

*It is possible to organize a course in more than one collection. The hierarchy of the collection management can be done partially by choosing the parent collection while installing. More easier movement of the nodes in the hierarchy will soon be added as an additional feature.*

## Creating A New Course from a Collection Manager

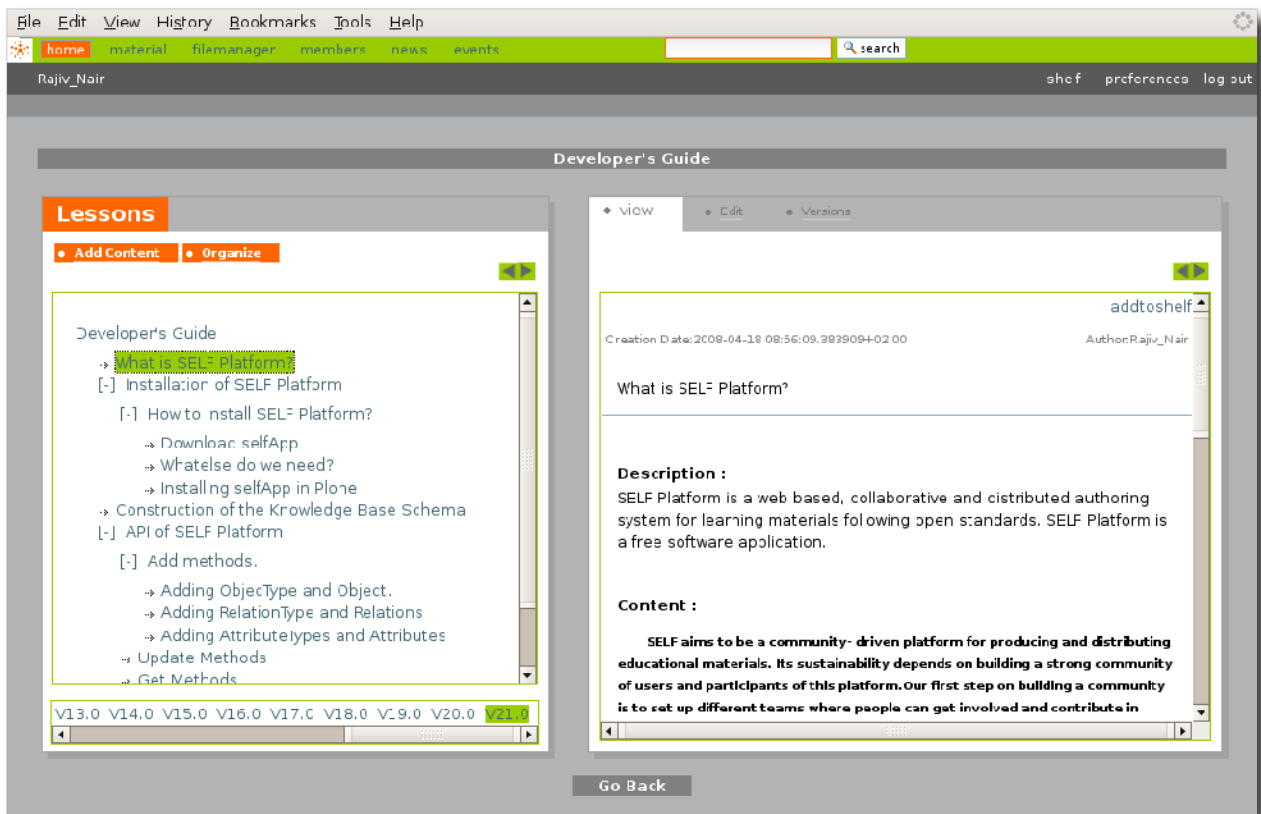


*Collection manager is also the place to initiate the creation of courses. Adding a course from a collection manager can be done by the action tab on the right side window.*

*While adding any learning object, the author can choose the language. The default language set currently is English, and eventually we will use the preference the author sets in the user preferences.*



# Course Manager

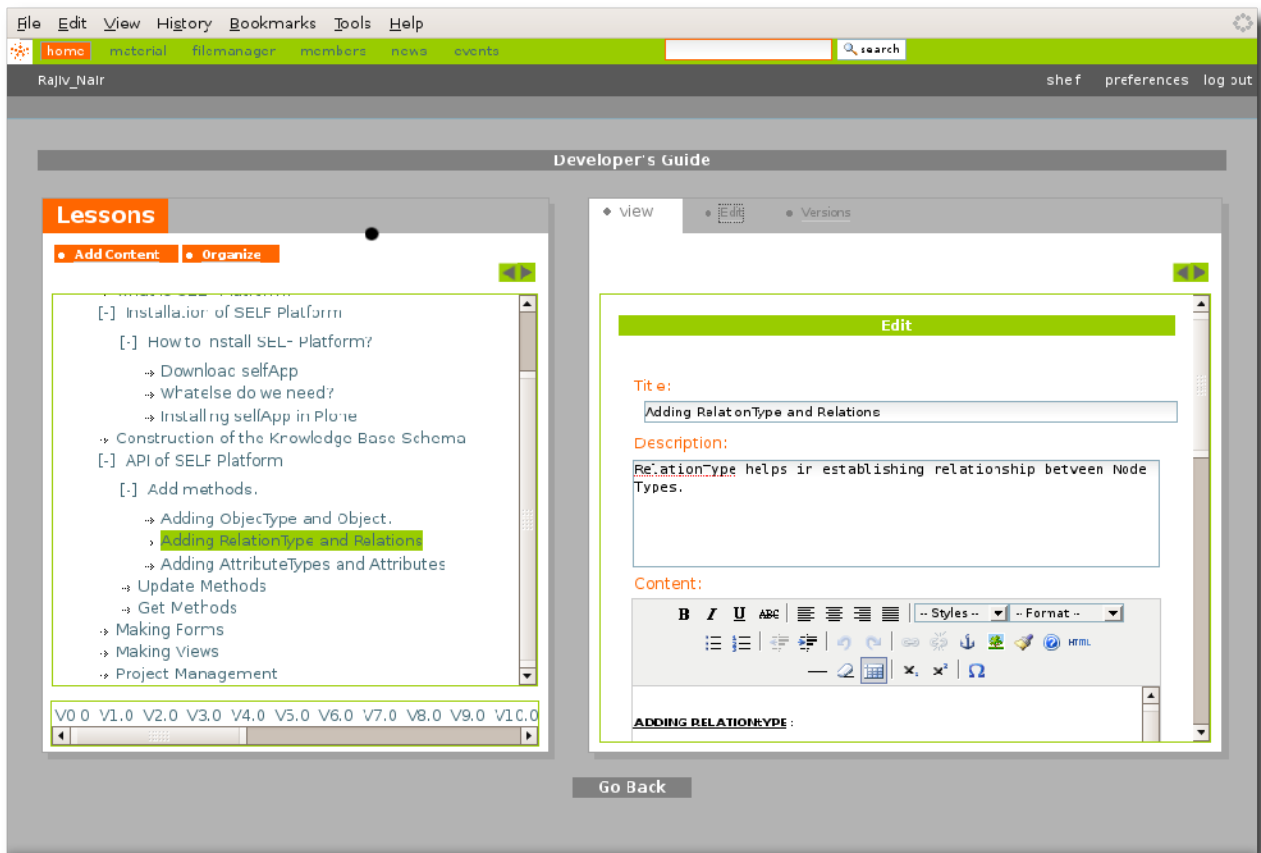


*The course manager is the core area where the action in the Platform happens. The course manager is designed in a way that a course is structured. Apart from authoring the content of the lessons, the focus of building the course consists in breaking the course into several activities, often nested in one another to create a hierarchically ordered structure. The Platform, uses a organizing and networking system, GNOWSYS, which allows apart from producing the hierachical organization, the activities can also be linked to other courses else where in the platform. Such combinations.*

*SELF Platform is unique for the feature of keeping snapshots of each structure of the course as it is evolving or as it is being modified. As hown in the bottom of the structure, the author can traceback the way the course is constructed.*

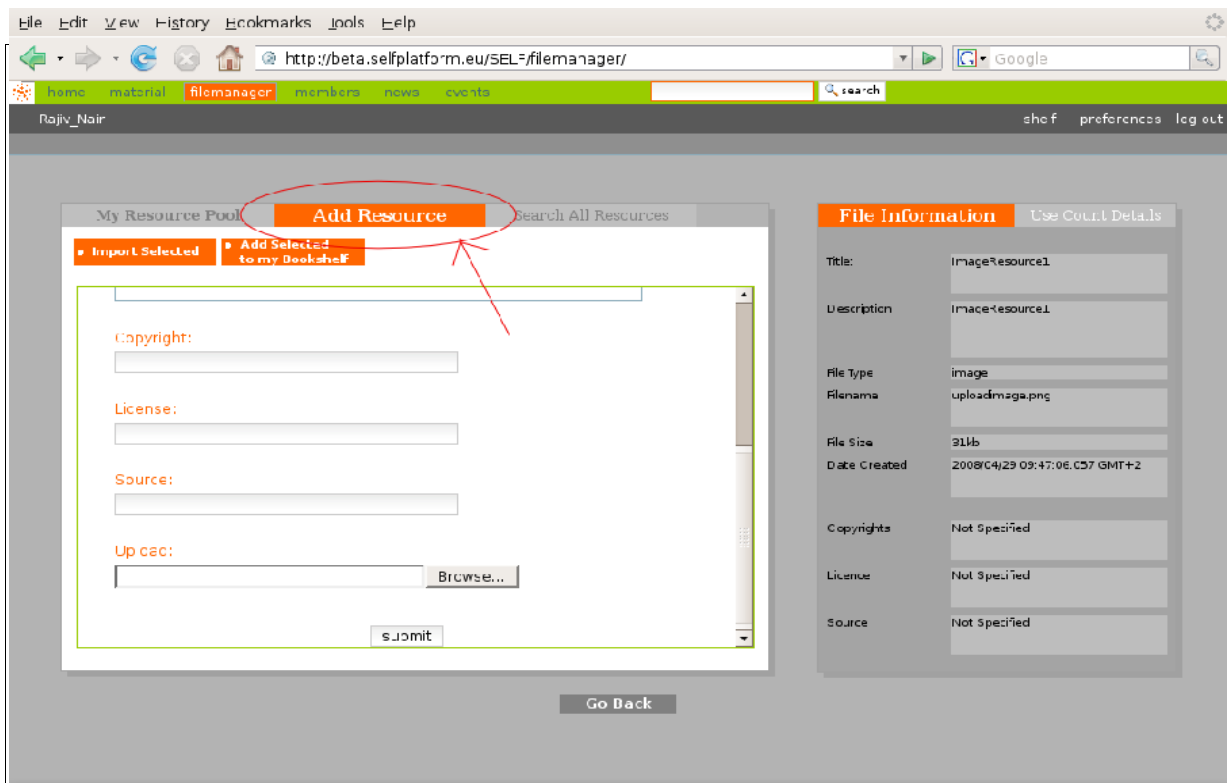
*There are several actions that the author can access from the course manager, the most often used being to add a lesson.*

## Editing a Lesson



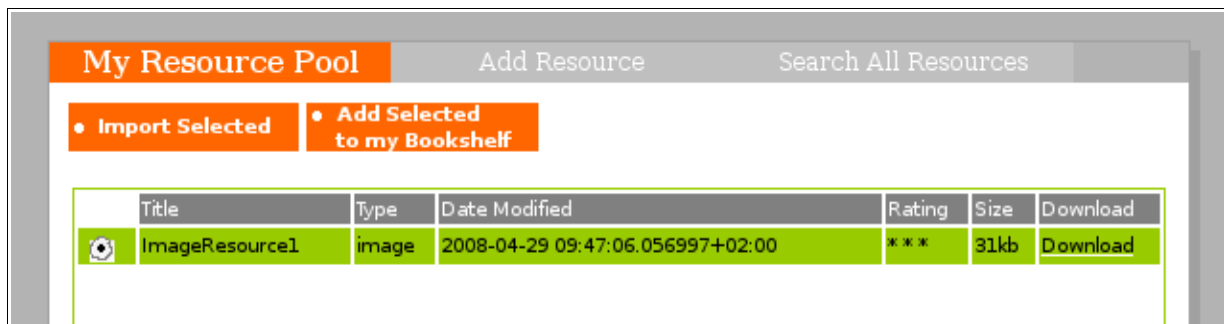
*Editing of a lesson can be done using a visual editor to create an HTML document. The editor is deliberately made smaller, for we do not expect the author to write an essay. This way the interface encourages the users to break the course into large number of learning objects (often called atoms or activities) with a single objective.*

## Uploading Images as Learning Objects



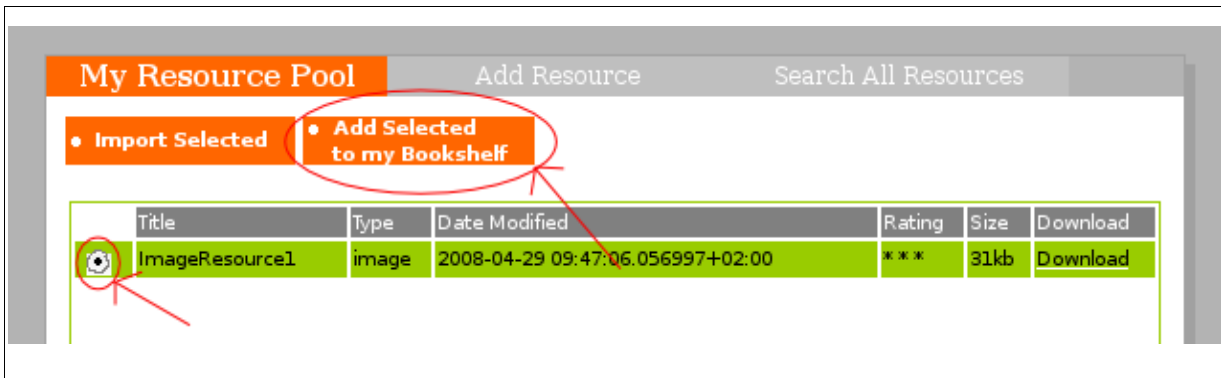
Uploading Images to the platform is done using the filemanager.

An Image is considered as a resource hence it is uploaded using the 'Add Resource' form in the filemanager.



The image uploaded gets added as a resource in the resource pool of the currently logged in user

## Adding an Image to the user's Shelf

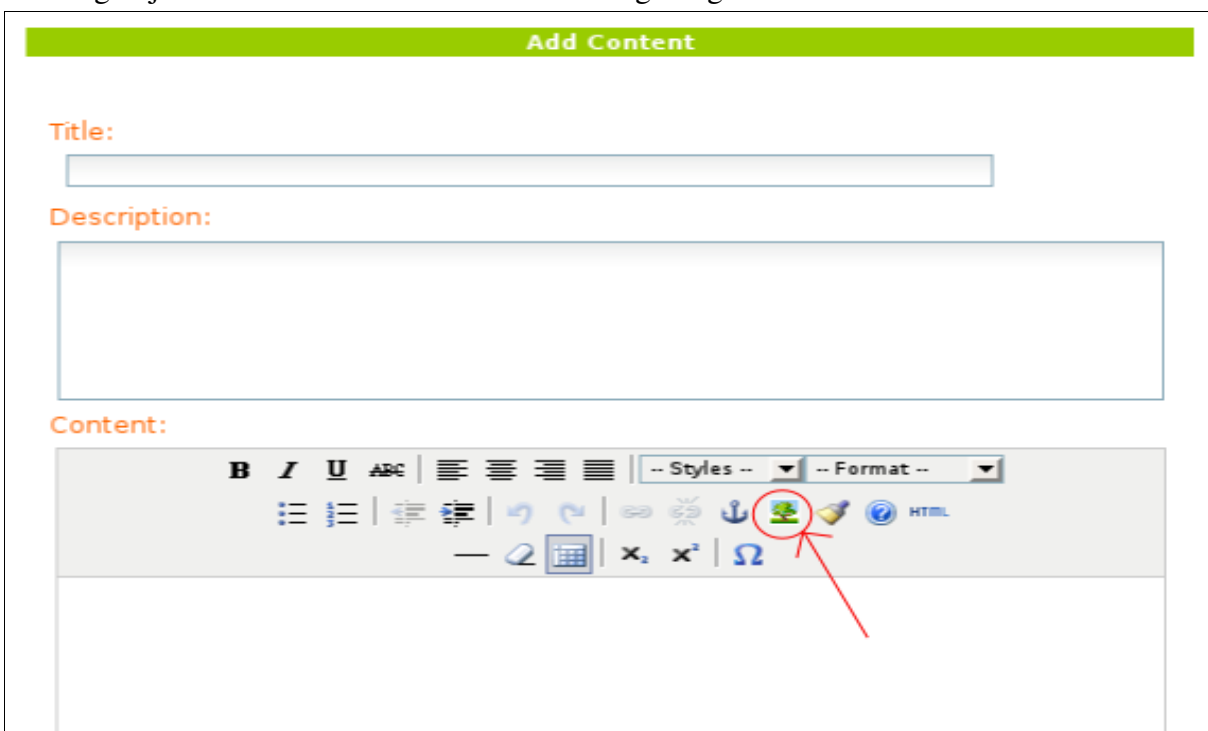


Adding an Image to the shelf makes the image added by the user accessible to rest of the users so that they may use them for constructing their courses or lessons.

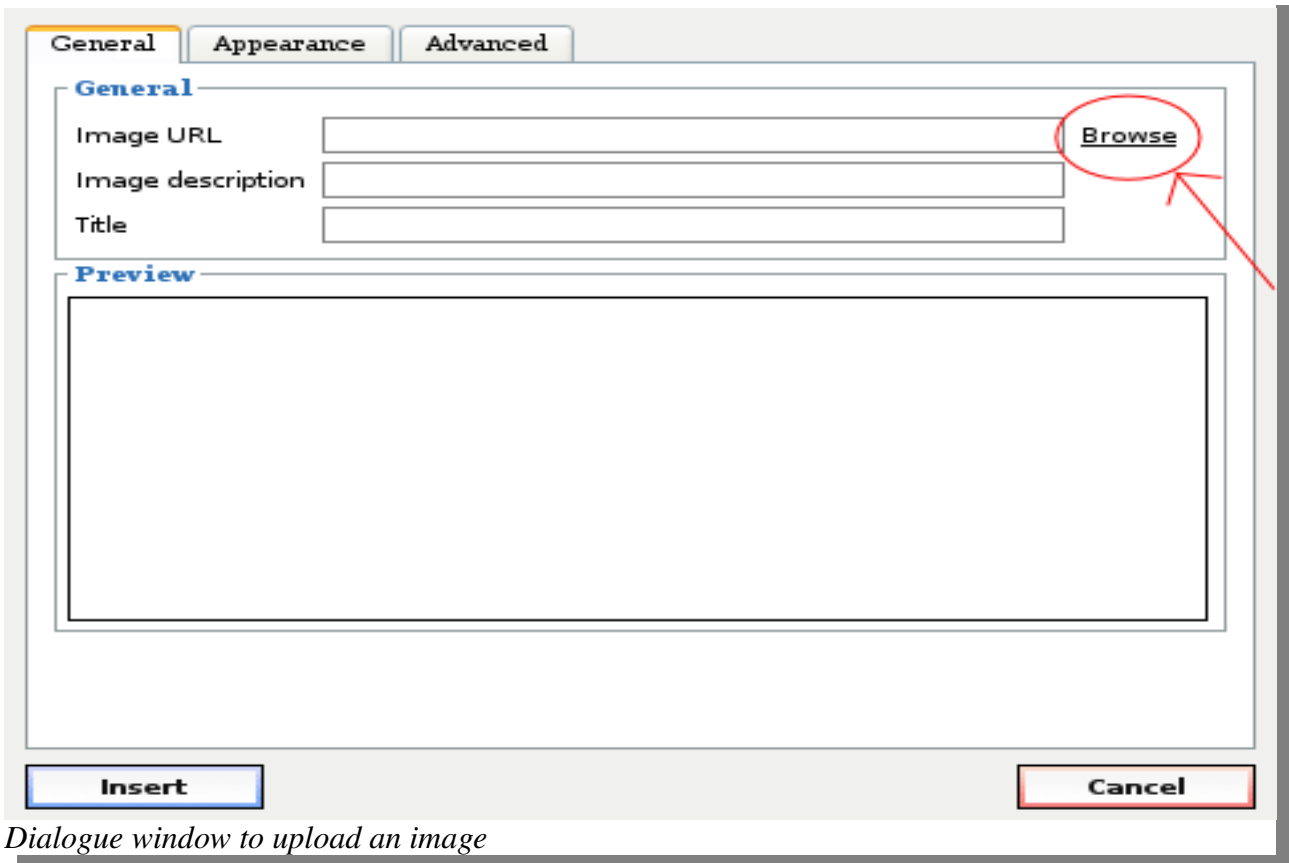
In short a particular user will be able to use all the images from his 'Resource Pool' as well as the images added by other users in their shelf.

## Using the Added Images

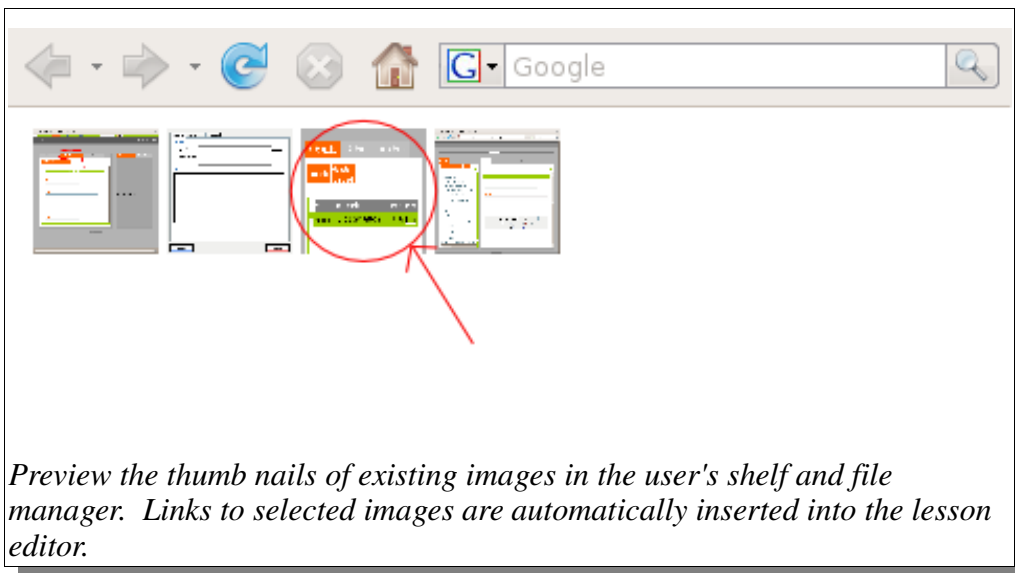
The added images in a lesson is initiated using the editor 'TinyMCE' used for creating and updating learning objects. The editor has a button for adding images .



Clicking on the button opens a window, where towards the extreme right there is a 'browse' button. Clicking on the browse button opens another window showing the images. The user can click on any image, add a description to it and then click 'Insert' to insert the image in the editor.

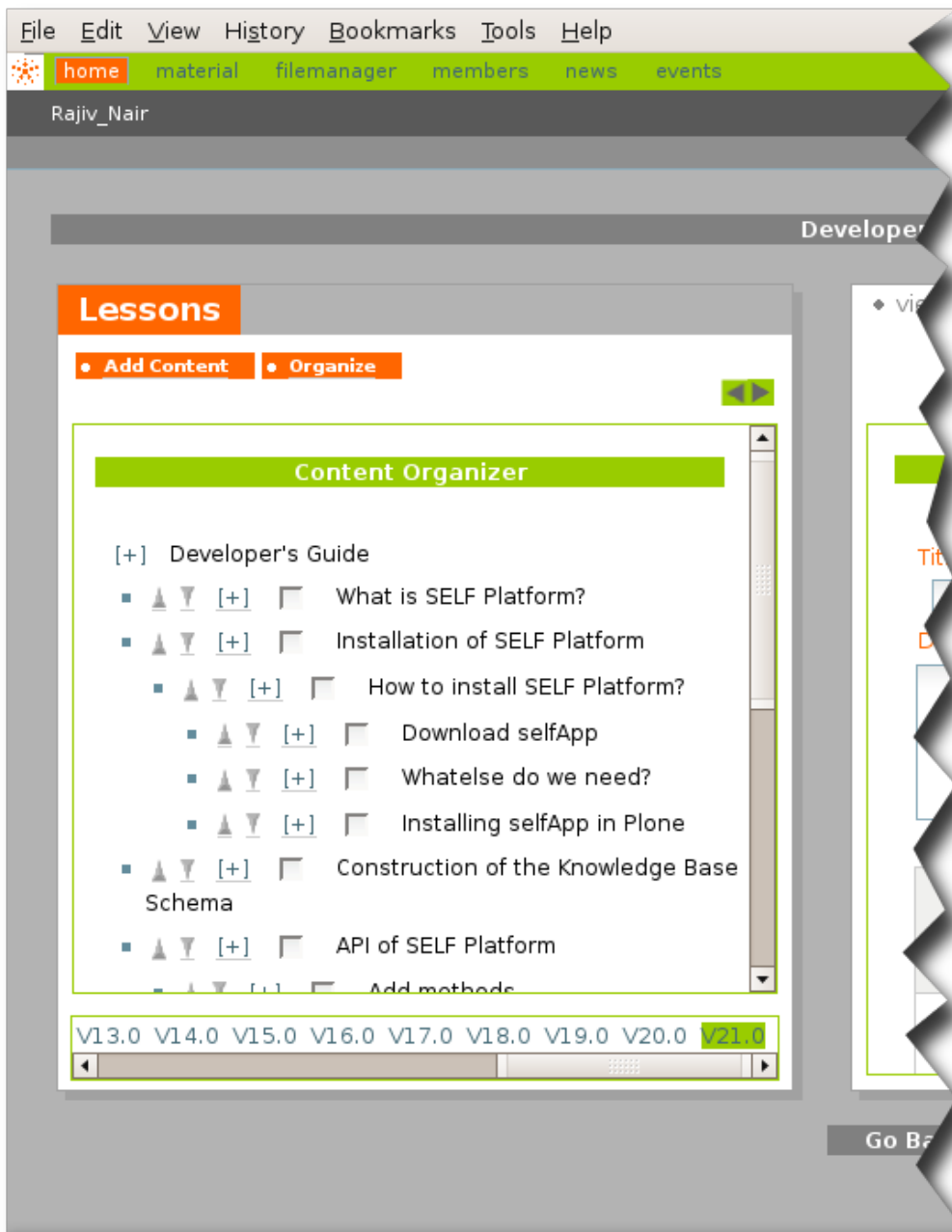


*Dialogue window to upload an image*



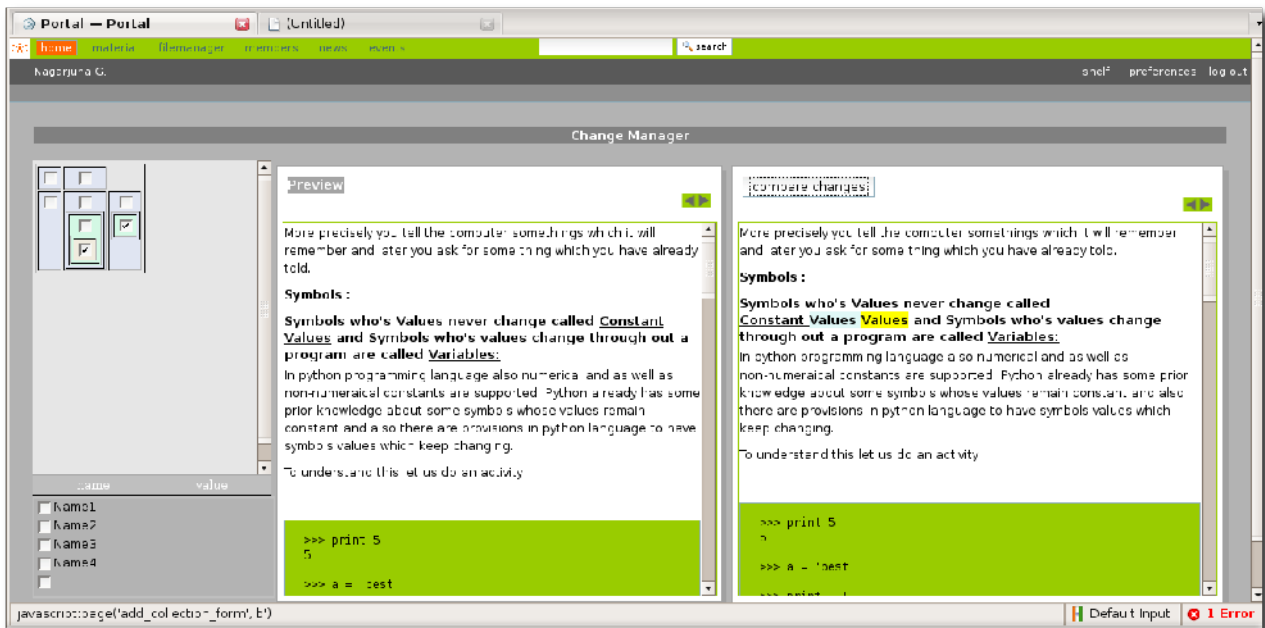
*Preview the thumb nails of existing images in the user's shelf and file manager. Links to selected images are automatically inserted into the lesson editor.*

## Course Organiser and Sequencer



*Organiser and sequencer of course components: it is normally preferred to deliver lessons in a sequence, one after the other. The lessons or activities delivered later are normally said to depend on the former. The platform offers the possibility to create the learning objects in any order, and then the resources can be reordered not only as a sequence of activities, but can be hierarchically organised. One or several objects can be selected and moved into a component of another complex node. Thus the Organiser helps in distinguishing learning objects that are leaves of a node and those into a node.*

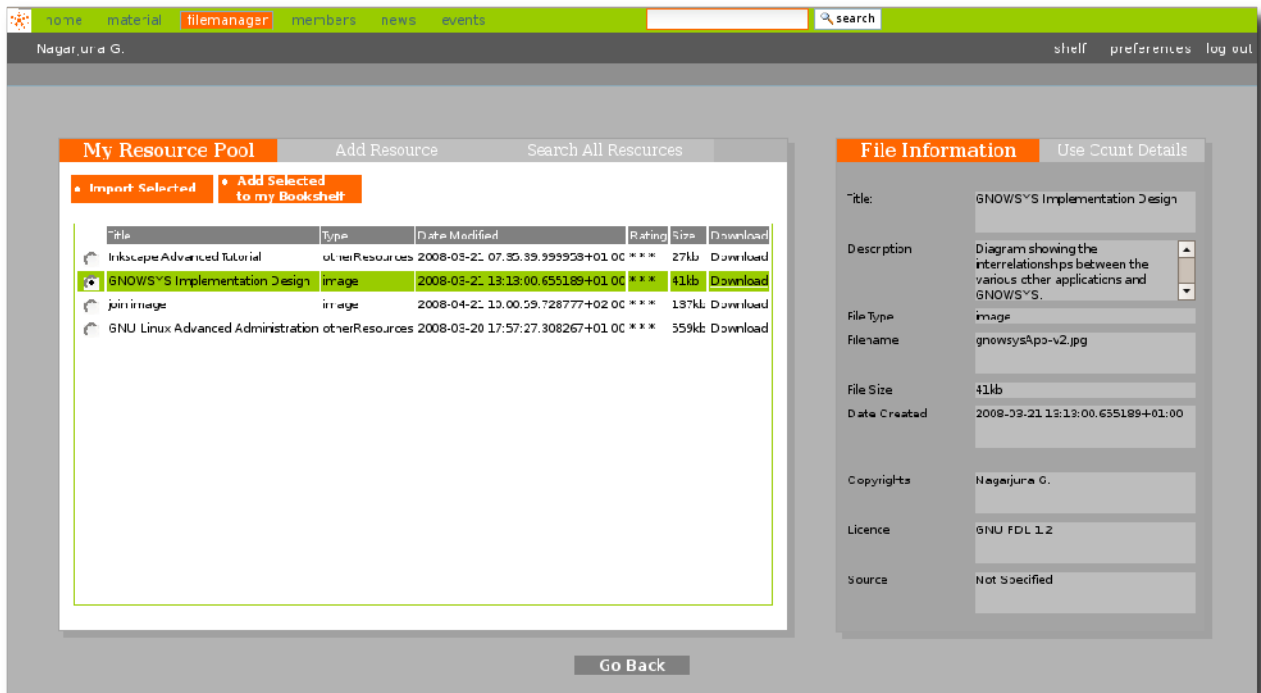
# Version Manager



*The version manager is a crucial tool without which collaborative authoring becomes very difficult. The current implementation provides some of the features and many more or in the development stage. The left window displays the history of the learning object as a tree indicating the linear and branching that often happens while collaborating working.*

*A selected nodes can be previewed and compared. The merged version can soon be saved to create a new learning object. This action will be soon implemented. At the time of writing this report, this was the last screen shot that we took, indicating that this is still a work in progress.*

## File Manager



The screenshot displays the File Manager interface. At the top, there is a navigation bar with links for 'home', 'material', 'filemanager', 'members', 'news', and 'events'. Below this, the user's name 'Nagarajura G.' is visible, along with 'shelf', 'preferences', and 'log out' options. The main content area is divided into two sections: 'My Resource Pool' and 'File Information'.

**My Resource Pool** section includes a search bar and two buttons: 'Import Selected' and 'Add Selected to my Bookshelf'. Below these is a table listing resources:

File	Type	Date Modified	Rating	Size	Download
Inkscape Advanced Tutorial	otherResources	2008-03-21 07:35:39.999999+01:00	***	27kb	Download
<b>GNOWSYS Implementation Design</b>	<b>image</b>	<b>2008-03-21 13:13:00.655189+01:00</b>	<b>***</b>	<b>41kb</b>	<b>Download</b>
join image	image	2008-04-21 10:00:59.728777+02:00	***	187kb	Download
GNU Linux Advanced Administration	otherResources	2008-03-20 17:57:27.308267+01:00	***	559kb	Download

**File Information** section shows details for the selected file 'GNOWSYS Implementation Design':

- Title: GNOWSYS Implementation Design
- Description: Diagram showing the interrelationships between the various other applications and GNOWSYS.
- File Type: image
- Filename: gnowsysAps-v2.jpg
- File Size: 41kb
- Date Created: 2008-03-21 13:13:00.655189+01:00
- Copyrights: Nagarajura G.
- Licence: GNU FDL 1.2
- Source: Not Specified

A 'Go Back' button is located at the bottom of the interface.

The File Manager is a place where the user stores files of various kinds, which then can be used in the courses. It includes the files to be imported, images to be linked or used within the courses, or source files, reference resources that can be made part of the courses, etc. The description of each file is displayed on the information window on the right side of the selected item, which is also highlighted. This pool can be used to upload and download files that are useful by the authors while working. A feature is implemented in the file manager to prevent the same file (identified by the checksum of the file) from being uploaded twice even if the file has a different name. Selected files can also be added to the bookshelf or if the file is a SCORM file, it can be imported directly into the knowledge base as a course.

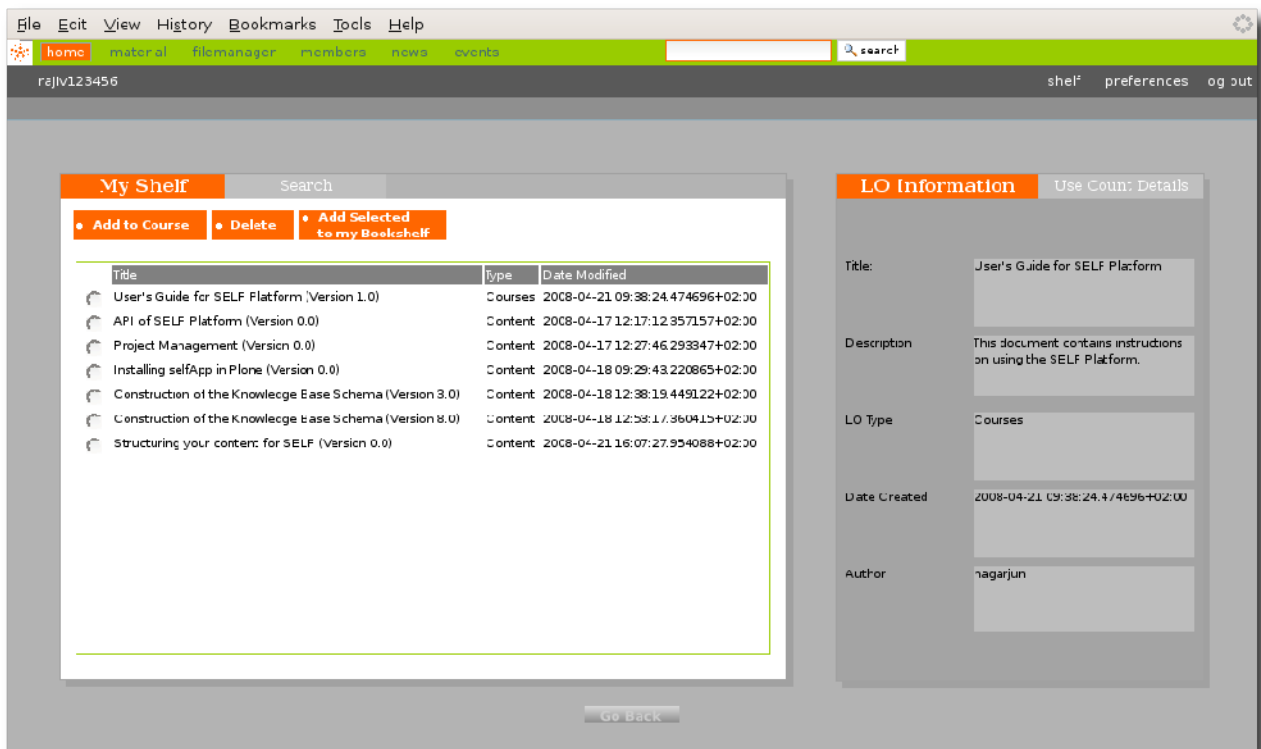
Images and other multi-media files can be either inserted or linked while editing lessons. This can only be done once such files are uploaded in the File Manager. Please see 'Adding an Image'.

Since over a period of time the resources can be so many, we display only the images that are uploaded by the user, and also those added to the shelf by the user from the over all pool of resources. The user can search resources by using the 'Search All Resources' tab and from there add the selected files to his/her bookshelf.

The selected files can also be deleted (the function is to be implemented very soon). When a user removes a file, it will only be eliminated from the knowledge base if the file is owned by the user and not in use by any other user, in case it is not, it will just be removed from the user's personal resource pool in the file manager and bookshelf. The removed files can be brought back by searching and adding to the pool again. The feature to permanently remove the resource will be reserved only if the files are uploaded by the same user, and if other nodes in the network do not have any relations with it.



# Shelf Manager



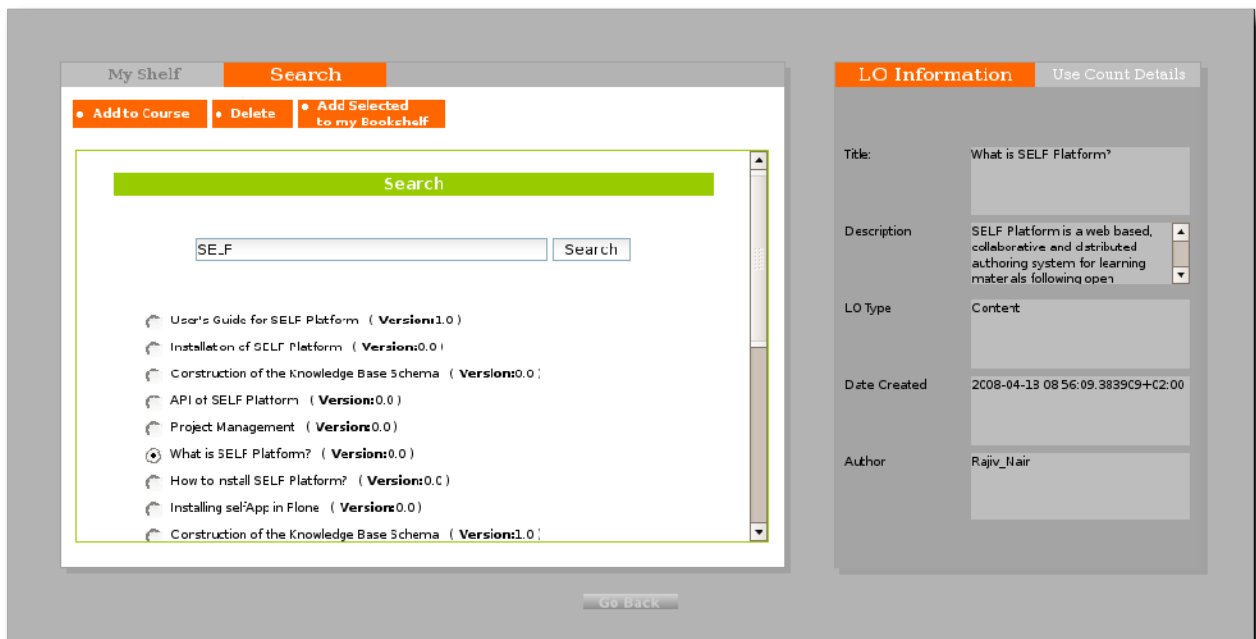
Shelf Manager is one of the unique features of SELF Platform. This does more than one thing in the Platform. Some of the things that can be done are:

- It is a place where interesting learning objects can be dropped or removed from the knowledge base
- User for rating authors as well as the courses they build
- It is a place from where one learning object from one course can be used in another.

A provision is also made to search for the resources from the shelf interface to that author can search and add them into the shelf.

•

## Search Resources in File Manager



*Each user uploads files that are useful and are not visible to the other users. Therefore it is very important to have a feature to search for resources that are uploaded by other users. The selected items from the search results can also be added to the shelf manager. By doing this, resources can be reused in other courses.*

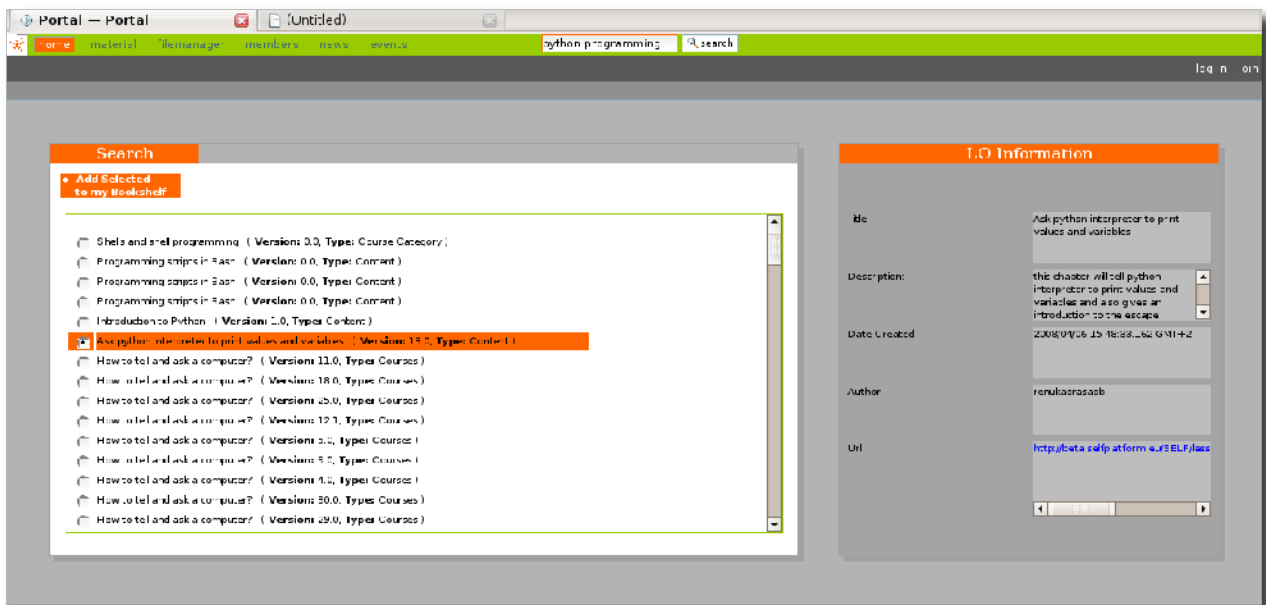
*A feature that is added to the uploading process is not to add the same resource second time by the users. Since authors are more likely to download and upload the free resources from the Internet, the chance of different users finding the same resource is very high. Therefore the filemanager logic computes for each resource a checksum (md5) and warns the users that the resource cannot be added, since it already exists.*

## Internationalization of the UI



*The Platform is made using the widely used inter-operable framework of internationalisation (i18n) for free software. The screenshot shows a partially internationalised form. As and when the translated messages are made available, the messages automatically appear in the default language set by the browser. SELF Project is committed to complete internationalisation, and will encourage participation from the community to get the platform translated in as many languages as possible.*

## Basic Search



A basic search of all the resources in the knowledge base is implemented. This feature currently uses only the title and the description of the resources. The result indicates whether the resource is a collection, course, lesson, image, etc.

### 4.4 Implementation of Features for Phase II

Most basic features are completed during the Phase I. The implementation of other remaining features, improving the already implemented features, bug fixing, more comprehensive documentation are the main engagements of the Phase II, which started from May 2008 till the end of the project in July 2007. The following are the list of features that were added in Phase II.

- Advanced metadata editing
- Complete Internationalisation framework
- Export to SCORM
- User Interface Improvements
- distribution and packaging of Software

### 4.5 Features for Phase III

The development of the platform will continue beyond the project period completes. The software development and maintenance of the project will be continued by the knowledge lab, and will be open to the developers from the free software community.

- SSL and Challenge Response Based Authentication
- Quality, Reputation and Popularity assurance mechanisms,
- p2p communication between platforms
- Exchange of documents in other standard formats: mediawiki, LaTeX, texinfo and DocBook
- Translation Interface

- Advanced Semantic Search
- distribution and packaging of Knowledge Base
- Atomiser
- Question banks according to QTI Specification
- p2p network to support Metadata Harvesting Protocol (OAI-MHP)
- Gnowser, a Gnome Desktop client application
- Learner's Interface for delivering the courses

## 5 Developers and Users Guide

The continuation and contribution of the SELF Platform by other laboratories any where in the world is not possible without documentation of how the platform is made and how it can be developed further.

A Developers Guide is already being authored by the developer team at the <http://beta.selfplatform.eu>

End user documentation is developed in line with the features on the Platform becoming available. Now that the platform has become more stable and feature rich, it is the time that interested users need basic user guides and documentation to make the necessary steps and get to use the Platform. This documentation is being developed on the SELF Platform itself and partially already includes information on how to use each of the implemented features as listed in section 2 of this document. It is our aim to work with the community and Learning Materials team to make available specific guides for different target groups and different activities, such as:

- About the SELF Project and SELF Platform
- Basic user guide to the SELF Platform
- Advanced user guide to the SELF Platform
- SELF Publishers: Editing guidelines on the SELF Platform
- License and legal aspects of the SELF Platform

Updated information can be found in the learning materials collection named “SELF”:

<http://beta.selfplatform.eu/SELF/collectionview/>

Note: URL's to the SELF Platform will in due course be changed to reflect the stable version and improved URL structuring schemes. In case of doubt the base URL <http://selfplatform.eu/> will always direct to the platform.