

# Project Manager

## WHY SCRIBUS NEEDS A PROJECT MANAGER?

“Scribus is an opensource DTP application which is primarily used to design small newspapers, brochures, newsletters, posters and books.” These lines-as seen on many sites like wikipedia- clearly emphasize the word “small” .

This is because presently Scribus cannot easily handle large documents with 100 or more pages – scrolling through pages gets tough, it takes much time to update pages and within that time scribus becomes inactive as theres is a lot of data to be loaded.

Anyways, Scribus users can still handle large Books and Newspapers by dividing a big document into small parts, (which are 20 – 40 pages documents)which can be easily handled for editing and combining them finally.

But the problem that arises is that when layout of the book needs to be edited, it must done multiple times for each document.(around 10 times for a 300 page book). And this problem is not small as there are many attributes in a document like master pages, layers, styles, images, tables, numbering variables, frames and scrapbooks which need to be edited regularly while designing a book.

So we need to solve this problem.

The Project Manager solves the problem by having a set of common resources(master pages, styles, fonts, images, layers, stored in shared location) for the documents. It would make accessing these resources faster and since only a fraction of memory is used now, the speed of application also increases. Updating the document would be so easy that with just one edit every document gets updated.

This would increase the preferability of Scribus over its other counterparts.

## PRECISE DESCRIPTION OF PROJECT MANAGER:

### IMPLEMENTATION OF BACK END :

Firstly Project Manager will start with creation of a new file called book.buk (.buk is a new filetype which will be added to Scribus.)

Secondly, project manager will be able to create new documents in the file book ,not more than 40 pages in size(i will call the collection of documents a “book” from now on). Preferably no. of pages=20 will be best.

Thirdly, it would have the ability to import existing documents and dividing it into documents of size between 20-40 pages.

From the imported documents the attributes will be extracted and then stored in the shared place in the book.buk. The attributes include masterpages, styles, images, tables, scrapbooks, frames, numbering variables etc.

Thirdly, project manager will have the ability to update, delete, create new attributes which when modified will automatically update all the documents in the book.

And most Importantly, the documents that are created or imported will not contain the shared attributes. Only the link or pointer to the .buk file will be stored in those documents.

Fourthly, the book will be able to synchronize those documents in proper order for numbering them, adding cover and back pages, making indexes and appendices for the final touch to the book.

Lastly, pdf generation will take place from the file book.buk which will be done by adding shared attributes to the documents and then printing them out.

### IMPLEMENTATION OF FRONT END :

Implementing the GUI will start after discussion with the developers ,community & ui\_iv\_iu team. It is very important to discuss it then propose a synthesis because there are already too many windows in Scribus which have their own structure. I will try to follow the rules which will make the structure of the new palette consistent (from objective\_vision\_cedric\_scribusui.pdf for dialogs vs palettes and other ui research).

After discussion, support for attributes like master pages, styles,tables,images, numbering variable and frames will be added to the main palette.

This would involve mainly the reading research papers developed by ui\_iv\_iu team and implementing them.

## TIMELINE

Discussion with the community and developers and their feedback will take place every week regardless of the work to be done.

April and May will be the learning period when I will be getting to know how will I implement the required deliverables.

April:

Discuss with the Scribus community how the user interface of Project Manager should be.

Getting to know more about Scribus codebase by solving more bugs related to styles and masterpages.

Last week: exams.

May:

Learning how sharing of resources is done.

Learning how new filetypes are made and the file handling with qt.

Learning how pdf generation takes place from scribus documents.

May 28-June 3:

Creation of the project manager file called .buk and its implementation of what files it would store.

June 4-June 10:

Developing classes for adding new documents to project manager as well as importing documents.

June 11-June 17:

Extraction of attributes from the imported documents and storing them as shared objects in the file storage location of .buk

June 18 -June 24:

Connecting the editing classes of the attributes(masterpages, styles etc) to the book manager after making them capable of editing shared resources(masterpages, styles etc)

June 25 – July 1:

Enabling the documents to read shared data from shared location and updating the pages in canvas whenever attributes modification has been done.

July 2 – July 7:

Implementing finer qualities of project manager, connecting all the above mentioned details, debugging and testing.

July 8 to July 10 :

Mid term evaluation preparation.

July 11- July 22:

Two weeks for designing the palette and making handles for accessing and editing all these attributes.Two weeks are sufficient because discussion and research would have been done in april and may.

July 23 -July 29:

Synchronizing the documents by putting them in order and adding numbering system to the project manager.

July 30 -August 5:

Efficient pdf generation of the pages(It may seem a bit difficult to carry it out in a week but I would have done the research work on this in MAY itself)

August 5-August 13:

Testing and Debugging of GUI and merging with the trunk(I would try merging code to trunk whenever some deliverable has been implemented)

August 13-onwards:

Suggested pencils down date – preparing final evaluation and giving a final touch to requirements.

## CONTINGENCY PLANS:

Should I be able to cover the above mentioned topic before the prescribed time, I will implement finer details of the project manager like having an indexing system, synchronizing other data, support for chapters and sections using text variables and little details like coverpage backpage, appendix etc.

## AFTER GSoC:

I would love to code for Scribus even after GSoC ends. I like the idea of contributing to a community and making its software better than its commercial counterparts.

I will add some more features to this project and make it better because I know this project is incomplete within the bounds of GSoC .I would solve the bugs and continue developing for Scribus like the guy who made tables successful in Scribus last GSoC.

## ABOUT ME:

I am pursuing btech from Dhirubhai Ambani Institute of Information and Communication Technology, India. I love reading books, pencil sketching and designing on pc.

Some of my friends in the designing field use Scribus and I got to know about this project through them.

## RELATED SKILLS:

I know C++ from my 11<sup>th</sup> grade and have been using it to make small applications and projects for school and college. I learnt qt about a couple of months ago for voluntary development of a contract bridge game(a multiplayer card game used on tablets and uses networking) which is a long term sponsored project. I have made a poker bot with my friends in java (uses oop) which can play with humans.

I solved a bug on the bug tracking system of scribus <http://bugs.scribus.net/view.php?id=10409>

I am currently improving on the import of master pages(like styles) because my project demands it. It will be complete in a couple of days.

I can use version control system very well.

Although I am a second yearite, I am highly inspired and motivated to develop Scribus and be a part of the larger community. I was browsing through the Scribus code and getting to know how things work in the code even during my mid term exams. Its feels good when you solve a bug after days of getting to know how palettes work through use of docs and how canvas paints and how much difficult it is to design a good plugin.

I know it would be a huge learning curve for me to code for such a successful open source community like Scribus and I would be find myself lucky to contribute even after GsoC.

## CONTACT INFORMATION:

Name:Rajat Gupta

irc : baazigar (irc.freenode)

Email: [rajarajatsraj@gmail.com](mailto:rajarajatsraj@gmail.com)  
[201001216@daiict.ac.in](mailto:201001216@daiict.ac.in)

facebook:facebook.com/rajarajat