During the course of this project, an Internet accessible database was created for the use of Dr. Brian Sturm. It will contain materials generated by students enrolled in the 'Principles and Techniques of Storytelling' class. These materials will be available to educators, librarians, and amateur storytellers at large. It is located online at http://www.ibiblio.org/storytime.

Headings:
Databases.
Computer-assisted Instruction.
Digital Libraries.
Educational Technology.
Storytelling.
STORYTIME: A STORYTELLER'S DATABASE
PROJECT REPORT

by
Sarah N. Kahn

A Master's paper submitted to the faculty of the School of Information and Library Science of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Master of Science in Library Science.

Chapel Hill, North Carolina
April, 2008

Approved by:

___________________________
Advisor
Table of Contents:

1. Introduction

2. System Requirements

3. System Construction

4. Conclusion

Introduction:

Dr. Brian Sturm teaches a class called 'Principles and Techniques of Storytelling' in the University of North Carolina at Chapel Hill School of Information and Library Science. During the course of this class, MLS students tell stories, and conduct preparatory research on the stories that they tell. They hand in their research in the form of a document called a 'cue card'. The cue card is essentially a detailed metadata record about each story and includes background bibliographic information, geographic origin, and published variants, along with information about the student's individual interpretation of performance: emotions they want to evoke, their target audience, how they will personally stretch or take risks as a storyteller, etc.

Dr. Sturm has a large collection of these cards that has evolved over the years. Since he is well-connected in the fields of storytelling and education, he wanted to make this resource available for public use; so he has created a static webpage with a flat list of the cue card files. They are currently in Microsoft Word format, on his work computer, but are also served from the static site. He would love for teachers and others to be able to contribute to the collection. He would also like for the HTML page to be more functional and better formatted. It would also be easier for his students if there was an online system that allowed them to submit their homework, rather than having to email Dr. Sturm's Microsoft Word template.
**System Requirements:**

The basic system requirements were: a) that it allow Dr. Sturm complete control over what content is and is not published to the Internet, regardless of what is submitted, b) that it allow students to submit their homework to Dr. Sturm via email, regardless of whether they wished to contribute it to the online collection, c) it had to allow unregistered Internet users to freely search and use the collection material. Other features that we have incorporated include a kid-friendly page with a simplified interface, and interactive access to storytelling video clips.

**System Construction:**

Stacey Lunden is a fellow SILS student, and we have taken several classes together and collaborated on group projects in the past. While taking Dr. Sturm's Storytelling class, Stacey approached me about the idea of collaborating on this project, and our work on the project has been documented on our Master's Project Blog at http://storytimemastersproject.blogspot.com/.

I immediately thought of using a content management system as opposed to building a system from scratch, for two reasons. A content management system would include a friendly administrative interface, which would allow the final product to be more useful to Dr. Sturm after we hand it off to him. A graphic administrative interface allows a user to make system changes and organize or update published content without any specific technical knowledge. Secondly, a content management system would have a robust user
registration/management module, allowing a backend administrator to easily keep track of who has logged into the system and submitted content. Additionally, neither one of us has substantial experience with MYSQL databases, and a content management system would eliminate that issue by building and managing the database for us. My experiences as a Research Assistant at ibiblio.org had made me familiar with the power and flexibility of many open source content management systems, so we never seriously considered any other option.

We began preliminary work on the project in November, 2007. We did a lot of research into currently popular content management systems (CMS), with the main requirements being that it was easily customizable, based on PHP, and that it was open sourced. We looked at Drupal, Joomla!, PHPBB, MediaWiki, Mambo, and BitWeaver. We ultimately chose Joomla! 1.5 for several reasons. It has a large user and developer communities, with hundreds of custom plugins, add-ons, and other extensions. It is also completely customizable with Cascading Stylesheets (CSS). Several of the other CMS's were table based and very square, and we wanted to be able to incorporate Joomla! into our CSS design, rather than design for Joomla!.

Early in the research process, we decided that the project itself would be hosted by ibiblio.org. Our hosting requirements were: a web directory and a MYSQL database. We also may require significant disk space in the future, depending on how the video collection grows. Fortunately, not only does ibiblio.org place no memory restrictions on
its users, it also makes supporting SILS projects a top priority.

We decided on a simple division of labor: Stacey would design the interface and collaborate with Dr. Sturm while I would build the back-end in Joomla!. Stacey created four separate designs with accompanying fairytale-themed art, and created prototypes for the more interactive elements such as submission forms. I took her designs and interpreted them inside of Joomla!.

The process of customizing Joomla! turned out to be more of a challenge than I had anticipated. One thing that I didn't realize initially was that Joomla! couldn't incorporate custom PHP (PHP Hypertext Preprocessor) without a third-party component designed to integrate it into the core code. I also didn't know that the back-end of a Joomla! template is somewhat counterintuitive in construction. One of the biggest challenges I ran up against in creating the custom templates for our design was incorporating Joomla! modules into Stacey's CSS. The main resources that I drew from in overcoming these challenges were the Joomla! Community Forum (http://forum.joomla.org/), and an excellent tutorial series by Barry North of Compass Designs (http://www.compassdesigns.net/).

Once I successfully migrated the four CSS designs into four templates, I began doing research on how to create custom forms inside Joomla!. I initially installed an open-source component called "ChronoForms" by ChronoEngine. But this was mainly
designed for smaller contact forms, and did not have the flexibility that we needed. Our forms needed to do three things: allow students to email their homework to Dr. Sturm, submit it to the collection if they so desire, and allow non-students to submit to the collection. After some more research, I came across a component called "Fabrik" (http://fabrikar.com/) which was more powerful and allowed for the creation of custom PHP database plugins. The only problem with Fabrik was that it was developed for an earlier version of Joomla!. We had a working installation of version 1.5, while Fabrik was designed for the 1.0.x series. Rolling back our installation would present a bit of a challenge, since I would have to completely re-work the templates. Version 1.0.x of Joomla! uses table based template design, and is less easy to customize with CSS.

Before making a decision, I created a test installation of Joomla 1.0.x as a proof-of-concept; which entailed setting up an extra web directory and database on ibiblio. Eventually, I did decide to delete our existing installation and roll back in order to use Fabrik. It was a good decision, since Fabrik has been able to do almost everything that we wanted.

First I will briefly explain the way that the system works, and then highlight specific details that involved customization.

1. Upon login, clicking the 'Submit a Story' button, a user is sent to the native Joomla! user login module. This page requires the user to register or enter an existing login and password. If they are new and must register, it sends them to a page which requests their first and last name, a valid email address, and their
preferred password. This is a key component of our security plan, since in order to activate their account, a user must receive the activation email at the email address they registered, and click the link. This ensures that a real person with a valid email address is gaining access to our data.

2. Once logged in, users are passed to a switchboard page which determines which form they should fill out. There are three separate forms, which are 90% identical, but do have slight variances in content and function. The forms are: non-student submits to database, student submits to database, and student emails only. We decided not to have a non-student email form, since it just doesn't make sense for unaffiliated users to send email to Dr. Sturm.

3. The user fills out the form, which does several things with the information. If the user desires, it emails them a copy of what they submitted, a copy is sent to Dr. Sturm, and if they wish to add to the collection, their record is submitted to the database, where it is invisible to the world until it is approved by Dr. Sturm.

4. Dr. Sturm receives an email telling him that there are new records, in addition to an email version of the record. He may then log in to the back-end and approve or delete the record.

5. Approved records are accessible via the 'Search' page, where I implemented a custom Joomla! mambot to allow users to search only records inside the Fabrik tables.

6. Records are also accessible via the 'For Kids' page, where a tabular view of the
database table displays selected fields, which are granted public read access.

Below are some specific features that we have designed:

- A PHP-driven form switchboard page. When a user registers and logs in, they are passed to this page to determine which form they should be directed to. Our original idea was to have one single form, and at the end of the form determine where the data would be passed, but this turned out to be impossible within the Joomla! framework.
● Back end administrator record control. There is a hidden "approved" field on student forms, which is accessible via the backend. An administrator may log in and view records which have been newly submitted, and determine whether they are suitable to be published out to the Internet. This way, Dr. Sturm will have the final say over what content is displayed.

● The web-forms themselves: Each form has a user receipt check-box, which harvests the logged in user's name and email address, and if checked, will send a
copy of the generated record to that email address.

**Conclusion:**

We hope that this system will meet Dr. Sturm's needs reasonably well. It would have taken far less time and effort to develop an equivalent system from scratch, but the powerful advantage of Joomla! is the graphic administrative interface. The graphic
interface will ensure that long after Stacey and I have graduated, Dr. Sturm or a student assistant can easily continue to administer without extensive technical knowledge.

We met briefly with Dr. Sturm prior to the formal demonstration of the website. His initial reaction was extremely positive. He has spoken about linking this website from his popular clips on Youtube.com, as well as linking it out to widely used teaching databases. There are some changes and improvements we would like to make in the future, but we are also confident that it will serve Dr. Sturm well as is for the foreseeable future.
Appendix: User Manual for Storytime: A Database for Storytellers

Table of Contents:

1. How to log in
2. User Accounts
3. Managing Records
4. Creating New Pages
5. Linking to the Menu
6. About Templates
1. How to log in:

The front end of the Storytime website is permanently located at www.ibiblio.org/storytime. To access the administrative backend, go to www.ibiblio.org/storytime/administrator. There is no link from the front end, since we didn't want users stumbling back there.

When you go to www.ibiblio.org/storytime/administrator, you will see this page:

Once you log in, you will see the administrative home page:
Things to notice here: the vast majority of the buttons you will not need to use, since most of the functionality of the website is achieved within the 'Fabrik' component, which is a third party add-on and not part of the dashboard menu. The small menu on the right shows you who is currently logged in. Along the top of the page, below the black bar, is the main menu bar. If you ever get lost, you can click 'Home' to get back to this main dashboard page.
2. User Accounts

When a new user comes to the site and registers for an account, the administrator will receive an email with the user's login information. To access a full list of registered users from the backend, in the main menu, click 'Site' then 'User Manager' as shown below:
The User Manager (displayed below) allows the administrator to edit user account information, or delete users. You can also search users.
3. Managing Records

The site uses a third party component called 'Fabrik' to manage the submit forms and database tables where the records are kept. To access Fabrik, select 'Components' from the main menu, then 'Fabrik', as shown below:

If you select 'Fabrik', you will get to a landing page, with the following options: Connections, Tables, Forms, Groups, Elements, Validations, Validation Rules.
While you will probably not need to do anything with most of these, here is a brief rundown of what they each do.

- Connections- tells the application which database to use. Our database is hosted at ibiblio.org, and the login
information is available in the Joomla! configuration file.

• Tables- allows for backend access to the data stored in the database tables associated with submission forms.

• Forms- are made up of groups of elements. On this page you can set access levels, set form functionality such as whether to email information or submit it to the database, and control which elements appear in any given form.

• Groups- are small groups of form elements. Having elements split into groups makes it easier to determine which elements are part of which form. On this page you can set the group name and the label that appears for end users, as well as add specialized css or javascript.

• Elements- are the smallest building blocks of the forms. Each input field is an element. In the elements page, you can add, delete, or change existing elements, as well as set permission levels, data validation rules, and type of input.

• Validations- shows you at a glance which validation rules are set for which elements. We currently have only one rule set up- whether the field is required or not.

• Validation Rules- allows you to set up different types of data validation checks. As previously mentioned, we only require that a field not be blank, but other types can be 'alphanumeric only' or 'valid email only', for example.
The Fabrik home page:

To access records in order to approve, add video, or delete, select 'Tables' from the Fabrik menu. In the image above, 'Tables' is highlighted. You can also select it from the Joomla! menu: 'Components' > 'Fabrik' > 'Tables'.
The Tables page:

Story records live in two database tables, 'Stories' is for student records, and 'Submit Non-Student' is for non-student records. They are separated because of the way the forms are set up. The administrator will receive a notification email regardless of which table a record is submitted to, which will specify which type of record it is. In order to approve a record, click 'View Database' next to the table you wish to access.
To edit a record, click 'Edit' in the far right column. This will take you to the editable version of the full record. The display is a streamlined version of the data entry form on the 'Submit Stories' page. The administrator is able to add or change any of the displayed data.
At the bottom of each record are the fields for approving the record to be publicly viewable/searchable, and the field for uploading video files.
When you are done making changes, simply click 'Send it!' to commit them. The process for approving/editing records for non-students is the same, with the exception that non-students have the option to add a video from the front end. The video name is a link, so to view it to make sure it's acceptable, all you have to do is click that link.

4. Creating New Pages

Just in case you ever want to add new contextual pages to the site... Joomla! is primarily designed to be a forum platform, so it has built-in features that allow for the management of pages composed of articles and user posts. It
also allows for static content pages, which is probably what you would want. To create a static page, navigate to Home > Static Content Manager (located on the icon dashboard.)

The Static Content Page:
To create a new page, click 'New' in the upper right hand corner. You may also edit, delete, or unpublish existing pages. The home 'About' page lives here, in case it ever needs to be edited/updated. Be sure to publish any new pages created, or they will not be visible to the www.
The actual create page template is shown above. It gives you the options to add a title, content, and set access permissions. Unless something needs to be password protected, the default settings of publicly accessible should be fine.

5. Linking to the Menu
The only menu that we are using in this site is the 'main menu'. There are several others available, and more can be created if needed, using the 'Menu Manager'. However, for the purposes of linking to our existing menu, we will focus on editing the main menu. From the Home page, select 'Menu' from the main menu, then mouse down to 'main menu'.
The main menu page: Here are all the links that are currently in operation on the site. You can create links to Fabrik components in this window, or link to other dynamic elements that Joomla! can create, such as RSS feeds. You can also link to articles and content pages which have been created by you. You may additionally edit, unpublish or delete existing links.

To create a new link, click 'New' in the upper right hand corner. This will send you to this page:
Let’s say that you want to create a link to a static content page which you have created. You would select ‘Link-Static Content’ from the ‘Content’ section:
Which would take you to this page: where you can create the link and define the parameters associated with it. You can set the name, which menu it belongs to, whether an image appears next to the link, etc.
6. About Templates

Templates are what define the look and feel of the website. Storytime uses four separate templates, which are each bundled into a package of files called Earth, Air, Wind, and Water. The template manager can be accessed from the Home page by clicking 'Site' on the main menu bar, and then 'Template Manager'. Select 'Site Templates' from the sub-menu. 'Administrator Templates' has to do with the look and feel of the back-end interface. 'Module Positions' has to do with the set up of the templates, so please do not touch.
Here is the 'Site Templates' page:
This page shows all of the installed templates. In order to assign a template to a new page, select the template you would like to assign by clicking the bubble next to the name, then click 'Assign' in the upper right hand corner.
The 'Assign' button will take you to this window, which lists all the existing Joomla! pages that are available.