Use-Cases

1) Add a file

* User selects "Import..." from options menu
* Application opens a file browser dialog box. User can enter file path name or browse to it's location and select by left click
* User selects whether they want to move the file to the LIBRARY or leave it in place (If moving to library application copies file to it's own directory system)
* The file is examined (with a MIME type utility library such as mime-util or JMimeMagic, see <http://www.rgagnon.com/javadetails/java-0487.html>) to determine file type
* The application determines whether this is a file that the application can read (initially just pdf)
* A new File class is created with the current file attributes
* An entry in the library table of the database is created using hibernate (has all metadata of file stored with it)
* The application prompts the user to tag the file
* Once the user is done tagging (see tagging use case for more explanation) the library is displayed with new file highlighted (temporarily)

2) Edit a tag

* User right clicks on tag
* Tag menu appears. This has "Edit", "View Documents" etc
* User selects Edit option
* Application fetches tag metadata from database
* Editing options are: “Delete”, “Make subset of another tag” etc

3) Create tag

* User selects "Create tag..."
* Application opens create dialog menu
* Users enters tag metadata (name, tags that it associated with, category it belongs to, etc.)
* Application creates new tag object, sets tag metadata, adds tag to database

4) View library

* In the file navigator panel (user can browser, search and open a file)

5) Tag a file

* User select one or multiple files in file navigator panel
* User right click on them and select “add tag…”
* An tag dialog window pops up
* User search tag by tag’s name or create a new tag
* User select one or multiple tags
* User press “tag…”
* Application tag the selected files with the selected tags
* User keep editing until it press “done…”
* Application close the dialog window

6) "Watch a directory" for added files (e.g. Downloads folder)

* User choose "Watch a directory" function button
* Program prompt a directory navigator
* User navigate through file system and choose a folder
* Program add a new listener to monitor that folder

7) Add a note

* User open and view a file
* User choose add a note
* A plain text editor window pops up and allow user to write comments
* User save the comments
* Program create a note associate with this file, this specific page and store it in database

8) Get info about a file

* User choose a file or not (by default the program will assume it is the one that is currently being viewed)
* User choose get info function button
* A window will show up with all the info about the chosen file (tag info, tag hierarchy, not, modified date, download date, view count…)

9) Get info about a tag

* User choose a file or not (by default the program will assume it is the one that is currently being viewed) or User choose to input the tag name by typing into a prompt window
* User choose get tag info function button
* A window will show up with all the info about the chosen tag (tag name, tag hierarchy, number of file with this tag, list of files with this tag…)

10) Text Search (not sure how to do this yet)

* User choose to do text search
* User type in text
* Program output entries of file and location of text

11) Search by tag (same as Filter by tag)

* User choose search by tag function
* User enter one or several tags name
* Program output list of files with relevant tag(s) order by reference

12) Read a pdf

* User double click a pdf file in file\_navigator panel or choose a pdf file and press open button
* Program open the pdf file in display panel

13) Bookmark a file

* User choose a file (by default the program will assume it is the one that is currently being viewed)
* User press bookmark button
* Program add the file into bookmark list, which is shown on bookmark bar
* User click one bookmark, the associated file will be open immediately in display panel