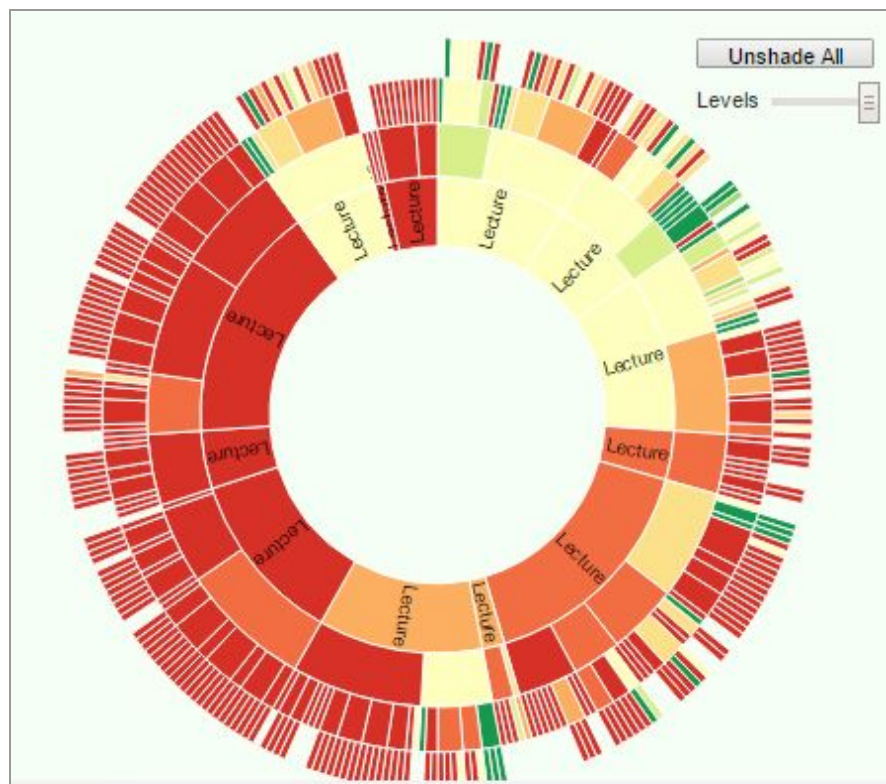# Reading Circle Final Report

Problem Statement

For INFSCI 2470 Interactive System Design we utilized the ReadingCircle web service to complete the assigned course readings. ReadingCircle is a social navigation and social comparison interface to engage students into online reading and facilitate the reading and navigation process. It was developed by the PAWS lab in the School of Information Science at the University of Pittsburgh. The programming was originally done by Julio Guerra, Denis Parra, and Peter Brusilovsky. Multiple courses in the program implement this system and use it to guide students in the completion of their literature. The system is functional, but not perfect. There have been various builds produced by the PAWS lab, and the one we use in class is a basic version. The user can read a generated jpg of a page out of the textbook, or complete the occasional sample question. The GUI still has some bugs, and the "Peer Comparison" section is incomplete.

Client

The client in this situation is our professor, Peter Brusilovsky. Since it is an application to assist students in the course, the person teaching the course has the most investment in the result. If the service is successful, the students of his course will benefit and more easily absorb the knowledge being taught in the curriculum.

User

The user in this situation is the student enrolled in the course. In order to be successful in Interactive System Design, one must stay on top of the assigned reading. The readings draw from multiple textbooks and sources, so it is beneficial to have a service that aggregates the various texts. The more efficiently the students can read the material, the better they will absorb it. The increased comprehension of the course concepts will only lead to higher scores on the in-class quizzes.

Primary User Persona Analysis

Age: The user could hypothetically be any age. However, if they are in graduate school there is a good chance they have just finished undergrad and are in their early 20s. We want to consider the average or majority of the people that will end up using the web service. For now, most of the users are young and still getting their education. Therefore, they are probably

acclimated with web service interfaces and would not have much trouble utilizing the ReadingCircle.

Education/Training: To be in a graduate program you have to have a degree already. This means most of our users will be intelligent and academically inclined. The degree could be in another field that is less technical than information science, but you do not have to be a programming genius just to use and benefit from the ReadingCircle application. However, if the user has an undergraduate degree in IS, CS, or another STEM major, they will probably be more than qualified to use and understand the functionality of the web service.

Class: Based on some of the previous analysis, we believe our users will not be complete novices with computers. However, there is always the possibility that they will not be experts. This means the interface can not be overly complicated or technical. Most of our users will have familiarity with a computer system at an intermediate level.

Demographics: The School of Information Science is especially diverse. There are a myriad of students from multiple ethnic backgrounds. We are proud of the international representation in the program, but it can also arise some complications in the interpretation of how to use the web service. We need to be sure that no design aspects rely on nuances in the English language, and the terminology is simple and understandable.

Goal: The goal of the user is to efficiently read the course material, keep track of progress, and use the absorbed knowledge to guide their study and be successful in the course. Having a better understanding of not only their personal progress, but the progress of the entire class, will help them to identify where they stand and how much more studying needs to be done to be successful. Books are large, heavy, and make it difficult to find a specific concept. Having an electronic interface can streamline access of the material and make it easier to learn.

Example Quote: "I just want an easy to use web service that makes reading the course material efficient, social, and informational"

Functionality

The web service is accessed through the Knowledge Tree interface, and is only accessible once logged in with a valid Pitt username and password. This allows on the initial load to pass in a handful of GET variables in the url. The username, group, and session ID are all able to be pulled from the url and used in the code. The username is the Pitt username the user entered to access Knowledge Tree. The group is the class the user selected from a dropdown menu to specify which class they are enrolled in. The session ID is a unique identifier assigned every time a student logs into the web service. It is used in facilitating the creation of temporary caches and logs, and signifies the single use of the ReadingCircle application. The graphical user interface is separated into two main sections. The left frame contains the circle

visualization to display the progress of the user's reading. Below the circle widget is an outline of all the available readings separated into sections. There is also a help link there which takes the user to a document outlining how to use ReadingCircle. This is useful help system that is readily available at all times. The right frame contains the actual reading material. Based on what link the user selects, a jpeg of the particular page is displayed for browsing.

Proposed Functions

We will facilitate the current reading circle system. After the interview with our clients, the system should be facilitated to meet the requirements listed in the table below. The code Rx means Requirement x.

Code	Name	Type
R1	The system should be able to compare questions and reading among all the other students (eg. individual students vs. entire class)	New function
R2	To make questions more visible	Improvement
R3	Student can provide feedback	New function
R4	To generate a new session ID every time user logs in	Improvement
R5	To cache the user information every time system request the model	Improvement
R6	To create a log to record everything user has been reading	Improvement
R7	Focus on the relationship among page, section and chapter and try to improve current design.	Improvement

Hierarchical Task Analysis

View Your Reading Progress:

1. Access "Reading Circle"
   a. Go to course site
   b. Enter "Knowledge Tree" portal
      i. Enter login information
      ii. Select "ISD IS 2470 Spring 2016" and hit submit
   c. Click on "Reading Circle" link

2. View your progress
    a. Click on "My Progress" tab
    b. Select which lecture you want to view results from
    c. Browse through report of your progress
3. Use appraisal of your progress to guide future study

Compare Your Reading Progress To Other Students:

1. Access "Reading Circle"
    a. Go to course site
    b. Enter "Knowledge Tree" portal
        i. Enter login information
        ii. Select "ISD IS 2470 Spring 2016" and hit submit
    c. Click on "Reading Circle" link
2. Compare your progress with others
    a. Click on "Peer Comparison" tab
    b. Select which lecture you want to view results from
    c. Browse through report of your progress compared to others
3. View statistics on overall class progress
    a. Asses the average score for the class on various exercises, and compare your scores to the mean
4. Use appraisal of other's progress to guide future study

Client Meeting

We coordinated a meeting with our client, Professor Peter Brusilovsky, to discuss the ReadingCircle application and where there is room for improvement. Julio Guerra, the project leader from the original development of the web service, was also present and provided valuable insight into the functionality. We learned that the first version of the system was developed three years ago with the goal of visualizing progress in the course reader. They used Javascript and PHP to implement their idea. It is self contained, and queries info using the Knowledge Tree portal. Based on the user's information, it chooses the specific class and compares progress to only other students from that class. JSON and XML files are used to ask the service for the progress of the class. There are also logs that store the progress of everything someone has been reading. To populate the circle widget, these logs are queried for their data and fill the visualization. Julio also differentiated between selecting a book, chapter, section, and page, and how there are different levels in the outline for each. It was critical that the design of the application is based on task analysis and what the user actually aims to accomplish by using the reading service.

Database Implementation

The data for the ReadingCircle application is stored in a large 42 megabyte mySQLi database. The database is then accessed by various php and javascript pages, which query the data to use. To function with the program, the database is hosted using the mySQLi server framework on an Apache Tomcat server. The application must also be hosted on the server in order to be accessible. This database contains 13 tables. It is important to have multiple tables to separate the data and link them in meaningful ways. Using these interconnected databases, a user persona is formed that allows the interface to be tailored to your progress. The thirteen schema names, and their contained columns, are shown below:

readingcircle_dev groups
- grp : varchar(50)
- courseid : int(12)

readingcircle_dev doc_lecture
- doclectureid : int(10)
- lectureid : int(10)
- docno : varchar(255)

readingcircle_dev answers
- idquestions : int(11)
- answer_num : int(11)
- answer_text : varchar(500)
- correct : tinyint(1)

readingcircle_dev course
- courseid : int(10)
- coursekey : varchar(20)
- title : varchar(250)
- domain : varchar(20)
- comment : varchar(500)
- legacy_kseadb : varchar(50)

readingcircle_dev tracking
- actionid : int(18)
- actiondate : datetime
- uid : varchar(50)
- grp : varchar(50)
- domain : varchar(50)
- sid : varchar(50)
- actionsrc : varchar(50)
- actiontype : varchar(50)
- docno : varchar(50)
- docsrc : varchar(50)
- filename : varchar(50)
- result : varchar(50)
- comment : varchar(512)

readingcircle_dev document
- docid : int(11)
- docsrc : varchar(50)
- docno : varchar(50)
- title : varchar(255)
- spage : smallint(8)
- epage : smallint(8)
- filenames : varchar(255)
- fileformat : varchar(20)

readingcircle_dev progress
- id : int(11)
- date : datetime
- usr : varchar(50)
- grp : varchar(50)
- sid : varchar(45)
- bookid : varchar(50)
- docno : varchar(50)
- page : varchar(50)
- question : tinyint(1)
- top : decimal(12,4)
- bottom : decimal(12,4)
- time : int(11)

readingcircle_dev lecture
- lectureid : int(10)
- courseid : varchar(10)
- title : varchar(250)
- objectives : varchar(2048)
- comment : varchar(500)

readingcircle_dev questions
- idquestions : int(11)
- docid : int(11)
- question : varchar(1000)
- answertype : varchar(45)

readingcircle_dev book
- bookid : int(10)
- bookkey : varchar(50)
- format : varchar(50)
- folder : varchar(50)
- title : varchar(255)
- authors : varchar(255)

readingcircle_dev precomputed_models
- modelid : int(12)
- uid : varchar(50)
- sid : varchar(50)
- courseid : int(12)
- lastaction : datetime
- computedon : datetime
- avg_progress : float(6,4)
- progress : varchar(2048)
- detailed_progress : text

readingcircle_dev precomputed_models_copy
- modelid : int(12)
- uid : varchar(50)
- sid : varchar(50)
- courseid : int(12)
- lastaction : datetime
- computedon : datetime
- avg_progress : float(6,4)
- progress : varchar(2048)
- detailed_progress : text

readingcircle_dev submitted_answers
- id : int(11)
- usr : varchar(50)
- grp : varchar(50)
- sid : varchar(45)
- idquestions : int(11)
- time : datetime
- answer : varchar(100)
- correct : tinyint(1)

Design Process

First, we considered our user study, task analysis, and client meeting. We wanted to be sure the functionality we implemented was driven by the takeaways from previous research. It was obvious that it had to be seamlessly integrated into the preexisting system. Any new additions

had to perpetuate the visual theme of the application, and be understandable by anyone regardless of technical knowledge. Any display of progress had to be able to use the preexisting schemas and data to fully function. Elaborating on our midterm design report, we decided to use progress bars to display the progress and comparison of reading completion. Progress bars are familiar to most users and are a simple, efficient way to display how close a user is to finishing a passage of reading. The progress bars query the database to obtain specific information for that session using javascript. An iFrame is created that houses the flexible progress bar widget. The information is stored in variables and passed to the progress bars which visualize progress based on numerical values. It also can determine the book, document, and page number based on GET variables passed into the progress bar script. Different types of progress bars are used in different situation. General percentages are used for overall assessment of reading. However, segmented progress bars are used when a ratio is evaluated. This adds multiple visual dimensions to how the data is displayed. To deal with granularity, a user can expand an individual lecture to see what various chapters of reading are assigned. A particular progress bar is then used for each of these chapters.

Obviously, this functionality is a vast improvement from the "under-development" alert. Even having the tab, without the functionality, was a distraction and demonstrated the incomplete implementation. The final version is also more visually appealing than our prototype design. Since it contains the percentage within the widget, it also displays more quantitative measures of reading progress than the main circle widget. In addition, being able to compare your progress with others now improves the social aspect of using ReadingCircle. Having this comparative data will help students juxtapose their efforts with other, and identify if they are behind or ahead in the curriculum.
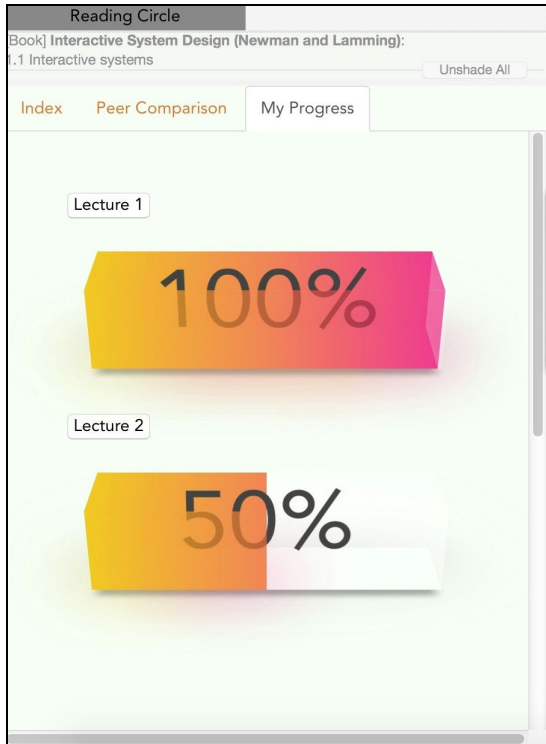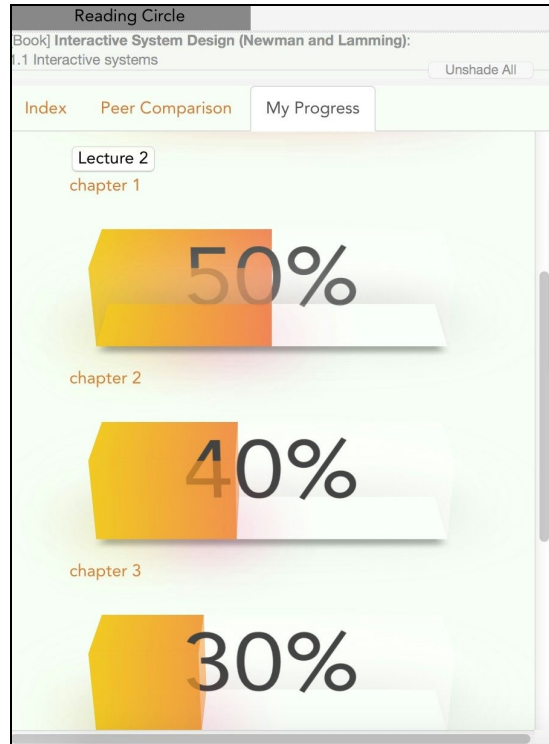
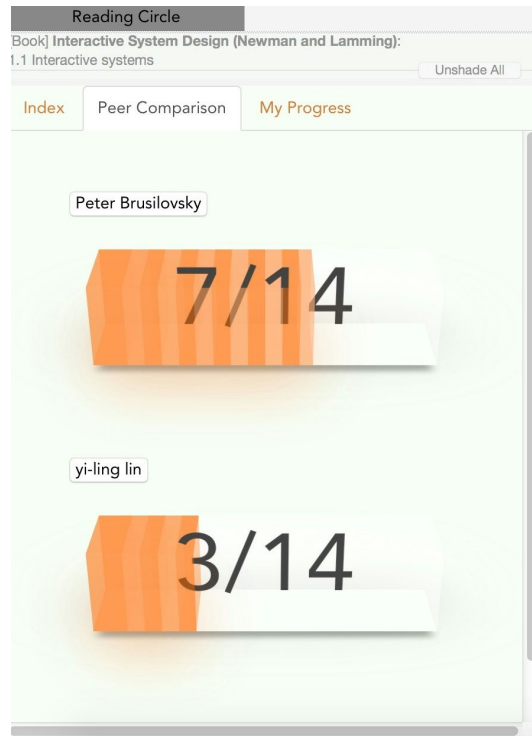**Old Design:**
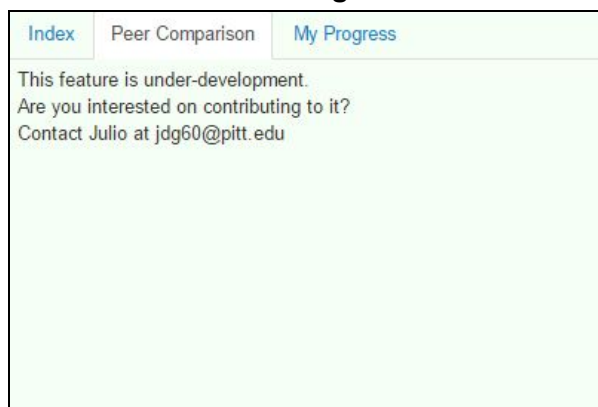
Prototype Design:
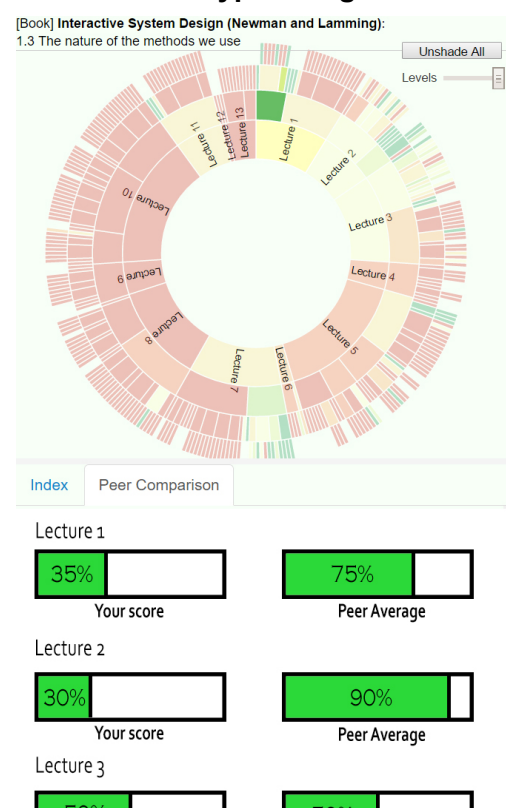
Final Functional Design:

Viewing your personal progress:

More in-depth detail of your personal progress:

Comparing your progress to others:

Book] **Interactive System Design (Newman and Lamming)**:
1.1 Interactive systems

Unshade All

Index   Peer Comparison   My Progress

Peter Brusilovsky

7/14

yi-ling lin

3/14

Video Demonstration:

https://www.youtube.com/watch?v=ayD9XrtL44g