

University of Alabama in Huntsville

LOUIS

Honors Capstone Projects and Theses

Honors College

4-21-2020

Primitives Needed for Online Peer Review Tool

Nicholas West

Follow this and additional works at: <https://louis.uah.edu/honors-capstones>

Recommended Citation

West, Nicholas, "Primitives Needed for Online Peer Review Tool" (2020). *Honors Capstone Projects and Theses*. 651.

<https://louis.uah.edu/honors-capstones/651>

This Thesis is brought to you for free and open access by the Honors College at LOUIS. It has been accepted for inclusion in Honors Capstone Projects and Theses by an authorized administrator of LOUIS.

Primitives Needed for Online Peer Review Tool

By

Nicholas West

An Honors Capstone

submitted in partial fulfillment of the requirements

for the Honors Diploma

to

The Honors College

of

The University of Alabama in Huntsville

4/21/2020

Honors Capstone Director: R. Kevin Preston

Lecturer of Computer Science

Student

Date

Director

Date

Department Chair

Date

Honors College Dean

Date



Honors College
Frank Franz Hall
+1 (256) 824-6450 (voice)
+1 (256) 824-7339 (fax)
honors@uah.edu

Honors Thesis Copyright Permission

This form must be signed by the student and submitted as a bound part of the thesis.

In presenting this thesis in partial fulfillment of the requirements for Honors Diploma or Certificate from The University of Alabama in Huntsville, I agree that the Library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by my advisor or, in his/her absence, by the Chair of the Department, Director of the Program, or the Dean of the Honors College. It is also understood that due recognition shall be given to me and to The University of Alabama in Huntsville in any scholarly use which may be made of any material in this thesis.

Student Name (printed)

Student Signature

Date

The purpose of this capstone, titled “Primitives needed for Online Code Review Tool” is to determine whether the implementation of an online peer review tool can be accomplished within the scope of a single semester by a Computer Science design team. The paper outlines the necessary steps required to investigate the feasibility of this project as well any implementation that shall be deemed necessary to pass along to the team assigned this project. As a result, this paper provides an overall set of requirements necessary to develop a prototype for this peer review tool. Provided are the different tools, software, and other techniques used in the implementation of the prototype. In addition, potential user stories as well as possible concerns have been provided as well.

In order to implement this online peer review tool, the program consisted of a client and a server. The server would be used to store a database, which contained information about reviewed items, as well as to store files that were being peer reviewed. The client would serve as an interface for the user to submit files and provide comments on files submitted for peer review. The client-server architecture makes the most logical since for this project because a peer review tool implies that multiple users must have access to the same shared data. The client can further be implemented as a desktop application that queries the server or a web interface that queries the server. For the scope of this project, it was required that client be a web interface.

Tool Name	Source	Purpose of Tool
NodeJS	https://nodejs.org/en/	Server-side Programming
VueJS	https://vuejs.org/	Client-Side Dynamic HTML Rendering

Bootstrap	https://getbootstrap.com/	Boilerplate HTML components
MongoDB	https://www.mongodb.com/	Database – NoSQL Implementation

Table 1 - Prototype Development Tools

Students have a good working knowledge of Java and C++. However, students may not be familiar with or even aware of the tools needed to implement this project. As a result, it may be time-effective, for the tool selection and analysis phase of the project, to recommend the tools listed in Table 1 to the senior design team. In the case the tools in Table 1 are not used by the senior design team, other alternatives have been provided. Due to the client-server nature of the project, Java will probably be the language of choice for most students as C++ is used more for embedded applications. If Java is used, Spring Boot would be the recommend framework as it automatically generates the code necessary to spin up a server as well as provide the capability that integrates with SQL databases and dynamically generate web interfaces. With the right skill set, the server could be written using C and the client could be written using custom HTML but this kind of implementation may be difficult to complete in the time span of a single semester. Another popular programming language common among students but not heavily taught at the university is Python. This could be potentially utilized by students as it is easy to use and pick up by beginners and has the necessary frameworks to create a web app in a timely fashion. Two of these frameworks include Django and Flask which provides the same functionality as the Spring Boot framework for Java that was previously mentioned above.

The criterial features that need to be implemented in this project that may present technical challenges are:

1. Ability to use the mouse to highlight a section of the document

2. Allow the user to enter a comment (text only) that is tied to the highlighted section
3. Save the comments and highlighted sections
4. Recall the saved comments and highlighted sections to re-display the document and comments in the web browser.
5. Ability to do the above steps with text or source files as well as pdf files.

Of the functionality above, Items 1-4 could be implemented with relative ease. Item 1 involved using simple built-in JavaScript functions to capture text highlighted by the user and triggering an event to generate a comment box. Item 2 consisted of creating a comment box component and associating that comment box with the line number highlighted in the web browser. Item 3 consisted of configuring the comment box to send the data provided by the user to the server by using a simple RESTful API. This API consisted of basic endpoints such as “<url>/comment/create” and “<url>/comment/update”. Item 4 involved retrieving the data from the REST API using simple HTTP response bodies and storing that data in a “comment” database that is tracked using the user’s name, the filename being used, and the line number of the comment. Item 5 involved creating a simple HTML page that reads in a file provided by the user and parses the file so that each line of the file can be assigned a line number. However, for context, it is important to know that I had prior experience with web development and the client-server architecture, so there was not as much technical debt involved when it came to these concepts. If this project was given to a group of students who have no web development or client-server experience it would be important for them to research these topics as this is a project that cannot be implemented as a standalone desktop application like other traditional senior design projects. In addition to the critical features in the list above, an additional feature was added for users to create a user account so that comments added to different files could be

more easily tracked and retrieved by users. Further details on implementation can be found in the repository provided with this paper.

One of the most difficult parts of this project was the ability to add comments to pdf files that were submitted for a peer review. Although there are many different libraries that deal with the manipulation of PDF files, many of them were either language specific or deprecated. Mozilla has created a library called PDFjs with the idea being a standardization of PDF manipulation across the web. Although the implementation of the PDF feature was not completed, it was primarily due to a lack of time. Despite this, the PDF feature can still be completed if given to a full Senior Design team.

In order to track my work and better simulate the circumstances in which the project would be completed, user stories and tasks were created to track progress of the peer review tool which can be seen in the figure below:

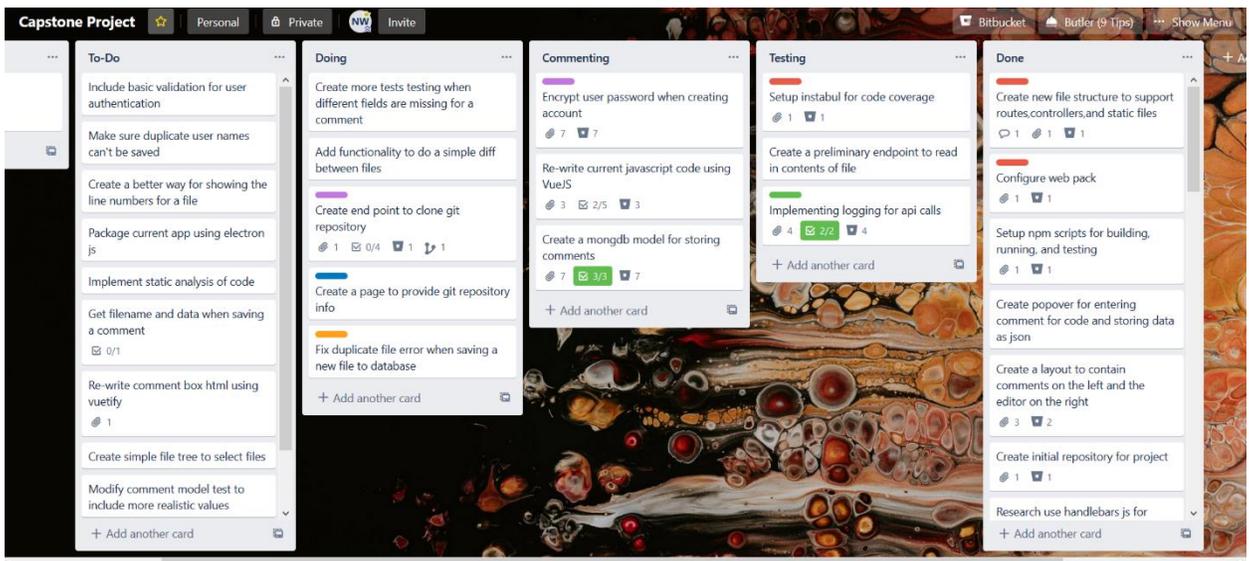


Figure 1

The full data will be provided with the paper.

In conclusion, if this peer review tool project was given to a Computer Science senior design team, it will be manageable to complete within a semester. Based on the fact that a majority of the features were able to be prototyped in a single semester by a single student, a senior design team should be able to implement the same features, including the ability to comment PDF files, as well as make it much more user friendly.

Re: Revised Capstone Report

Kevin Preston <rkp0001@uah.edu>

Tue, Apr 21, 2020 at 6:28 PM

To: Nicholas West <ndw0009@uah.edu>

Cc: "Dr. Ranganath" <ranganat@cs.uah.edu>, William Wilkerson <wilkerw@uah.edu>, David Cook <dac0010@uah.edu>

Nicholas,

I approve the attached Capstone Project.

Please follow the remaining steps from David Cook's email for the final submission.

R. Kevin Preston
University of Alabama in Huntsville
Computer Science Department - Lecturer

On Tue, Apr 21, 2020 at 6:06 PM Nicholas West <ndw0009@uah.edu> wrote:

On Tue, Apr 21, 2020 at 5:13 PM Kevin Preston <rkp0001@uah.edu> wrote:

Few things left.

1. My Title is Lecturer Computer Science
2. Delete all the comments

Make those changes and I will forward to the next group of approvers.

R. Kevin Preston
University of Alabama in Huntsville
Computer Science Department - Lecturer



Virus-free. www.avg.com

On Tue, Apr 21, 2020 at 3:52 PM Nicholas West <ndw0009@uah.edu> wrote:

I've attached the final version of the paper and the raw data from the tasks/user stories that I had put together.

Nick

On Mon, Apr 20, 2020 at 5:42 PM Kevin Preston <rkp0001@uah.edu> wrote:

Nicholas,

The next time your send to me include:

1. Once your manuscript is in final form, email the completed Capstone manuscript to the Project Director, along with the Title Page/Copyright Permission form.

See the attached for the forms to include.

R. Kevin Preston
University of Alabama in Huntsville



Virus-free. www.avg.com

On Mon, Apr 20, 2020 at 4:32 PM Nicholas West <ndw0009@uah.edu> wrote:
Attached is a revised copy of my capstone report.