Proctor Wiz

by

Cole Bowling & Ethan McMellan

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

Ethan McMellan	04/26/2021
Ethan McMellan	Date
Cole Bowling	04/26/2021
Cole Bowling	Date
Abdou Fall	04/26/2021
Abdou Fall, Faculty Advisor	Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services

April 2021

Table of Contents

ABSTRACT

Proctor Wiz is the perfect blend of functionality and security. This balance is not often seen in the proctoring service market, in many cases either the functionality is limited, and the security is sound, or the features are rich, but everything feels a little invasive. These invasive practices include requiring full control over the computer and tracking the students every movement. Proctor Wiz aims to fix the issues we see in current offerings, by offering essential features and the security that matters. Our product lets the user review any information taken from them to ensure that they feel safe, while also giving the reviewer an easy-to-use management and filing interface.

PROBLEM STATEMENT

Introduction

With more and more universities moving to online learning and the growing worries about personal privacy. There are very few services in the proctoring field that do not exhibit one of the following issues. Either they highly invade the students' privacy, process students' data, or invade their private life. That's why we created Proctor Wiz, a service that allows for both academic integrity and personal privacy so that both parties can be happy and satisfied.

Problem

Teachers need a way to proctor students when they are completing assignments, now more than ever due to mostly online schooling. Students across the country are having to use proctoring services that do a lot more than just watch your webcam and record your microphone. All the proctoring services out there are super intrusive and practically malware, recording the traffic that goes through your network, recording your webcam, your microphone, and your keypresses. Looking at the FAQ page for proctoring services such as Honorlock we can see mention of their "Search and Destroy" features and their ability to "Detect the Use of Mobile Devices" allowing them to track mobile devices on the network without the user knowing.

Solution

We created a proctoring service/application that properly manages and does the functions that one would want from a proctor while also eliminating the invasive functions that most proctoring services contain. One of the main ways we will avoid getting grouped up with competitors would be to have a very open and straight forward data collection policy and be transparent with what we are going to do when proctoring. We will avoid doing things such as tracking, and screen recording other devices that are on the network without consent of users.

Project Objective and Goals

- Record webcam, Microphone, and Screen and any combination of these three that the teacher requires.
- Store this data securely so that it can be viewed later.
- Automatically delete data after set time.
- Allow for student to review content before uploading.

Overview

The remainder of this final report outlines in detail how the project was completed. The report includes the following sections: Design Objectives, Budget, Timeline, Problems Encountered, and Future Recommendations.

DISCUSSION

Project Concept

We took this idea from our personal experiences with proctoring services. With new restrictions on how we go to school, online schooling has become an integral part of our lives. As it became more apparent that we would be doing all class work including exams online, we started to investigate the products our professors were planning to have us use. This led to a huge concern with how these programs were handling our privacy, and eventually led us to come up with Proctor Wiz.

Design Objectives

- Include a smart, easy to use interface that allows teachers and students to use the product efficiently
    - Visually pleasing design
    - Straightforward pathing through website
    - Different interface for teachers and students because needs are different for each user
- Include straightforward security measures to allow students and teachers to feel safe and protected while using the application
- Set up secure storage to house data collected from students, and show it to teachers in a way that makes sense and is secure

Methodology

Design Requirements

- These objectives allowed us to create a website that was not only visually appealing but easy to use. We tested multiple different paths through Proctor Wiz and found one that met in the middle of visual and user bliss.

Procedures

- During each phase of design, we stopped to look at what we had created and how it could be improved, whether that be through changing the interface or adding a descriptive box. This allowed us to hone in on how we wanted Proctor Wiz to work.

User Profile

These User Profiles helped us tremendously when deciding what kind of features to offer through our product. Form 1 is a teacher type person, while Form 2 is a student.

User Profile Form 1
Application: Proctor Wiz website/application.
Potential Users: Teachers hoping to proctor an assignment.
Software and Interface Experience: The user should be familiar with web applications and browsing the internet.
Experience with Similar Applications: Google Chrome, Firefox, Microsoft Edge, Safari, Facebook, Google Extensions.
Task Experience: Teaching a class online – Giving online tests / exams.
Frequency of Use: Will be used when the user needs to give a proctored exam or assignment. Used after proctoring to review students' information.
Key Interface Design Requirements that the Profile Suggests: The website needs to be easy to use and flows in a way that allows the user to be in and out quickly.

User Profile Form 2

Application:
Proctor Wiz website/application.

Potential Users:
Students taking a proctored exam.

Software and Interface Experience:
The user should be familiar with web applications and browsing the internet.

Experience with Similar Applications:
Google Chrome, Firefox, Microsoft Edge, Safari, Facebook, Google Extensions.

Task Experience:
Taking an exam.
Being proctored using a webcam and screen recording.

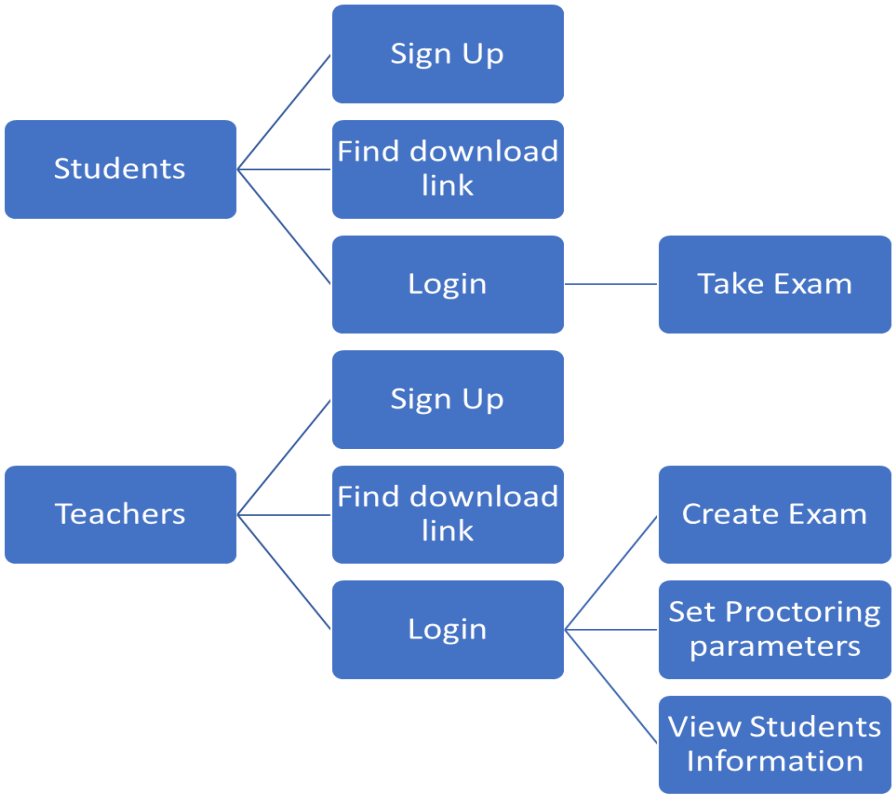
Frequency of Use:
When the user needs to take a teacher scheduled proctored exam.

Key Interface Design Requirements that the Profile Suggests:
The website needs to be secure, easily accessible to the user, and easily navigated by the user.

Use Case Diagram

This is an example of how each user would use Proctor Wiz. It goes through every option they have available to them.

Use Case Diagram

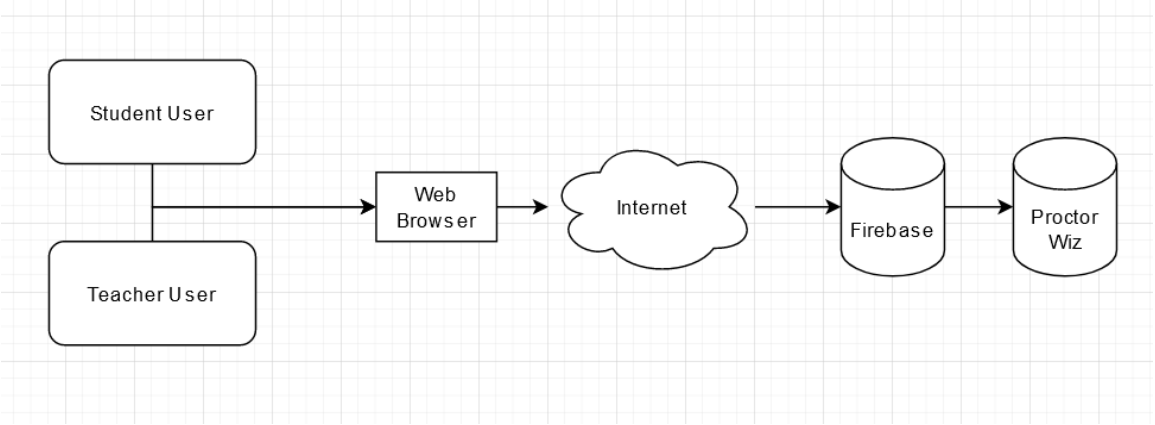


Technical Architecture

Website

Proctor Wiz is a web app that can be accessed from any device connected to the internet. The website is hosted on Firebase and is accessible publicly from the internet. Source code for the project is managed through a shared GitHub repository. This allows the Proctor Wiz website to be managed and changed from anywhere around the world. The Proctor Wiz website allows Students and Teachers to login, add/view/change assignments, and take assignments.

Technical Architecture Diagram



Testing

Testing Overview

To assess whether our product could be used effectively by end users. We had to test our system to see how it performed under load and if this performance fell under what we deemed acceptable conditions.

In this section we outline our testing conditions and the results of those tests.

Testing Methodology

We chose to go with a Systematic approach to testing. We believed this would best highlight any areas we messed up, this also matched how we approached our project.

Scope of Testing

1. Full Student workflow
    a. Signing up
    b. Recording videos
    c. Submitting Videos
2. Full Teacher workflow
    a. Signing up
    b. Watching videos
    c. Creating Assignments
    d. Adding Classes

Objectives

To have a coherent flow so that both students and teachers can progress smoothly from the student taking and recording their test to the teacher being able to then view the students' video.

A. Address all errors before Expo

B. Ensure smooth workflow

C. Test all users' cases

Test Logs and Procedures

Record #	Test case #	Input	Expected output	Actual output	Pass / Fail	Reason for failure/success	Date
1	1A	Cole	Video ask for permission to use camera, webcam, mic.	It asks you to provide permission and ask which device for each section if there is more than one option	P	It does what it needs to do and then also adds more to it.	01/05/2021
2	1B	Ethan	If you decline to give permission an error message appears.	If you decline to give access to a device, then an error message pops up that indicates that you will not be able to move forward with the process	P	This not only gives you an error message it also indicates that you will need to reload if you would like to move forward with the recording.	01/05/2021
3	2	Ethan	User Creates an account and we can view it through Firebase	This works we can see people that enter a correct email address	P		02/02/2021
4	3	Cole	Student user can review content	After the video has been recorded and user has	P	This works as intended as another tab will open and	02/06/2021

				clicked ‘stop recording’ a button will appear asking them if they would like to review.		allow the user to review the video.	
5	4	Cole	Student video is uploaded/downloaded (depending on if hosted or not)	After the video is done recording the option will pop up next to the ‘review’ button	P	This is currently working as intended.	03/09/2021
6	5	Cole	Teacher has the option to create assignment	There is a button on the teacher’s side that allows them to create assignments with different requirements.	P	The teacher has different options such as date range and front-facing camera or not.	03/12/2021
7	6	Cole	Teacher can create class with the option to browse file explorer to find roster list.	The option to browse is working and is pulled from the bootstrap documentation on how to get into the file explorer	P	The teacher can browse file explorer.	03/17/2021

Review:

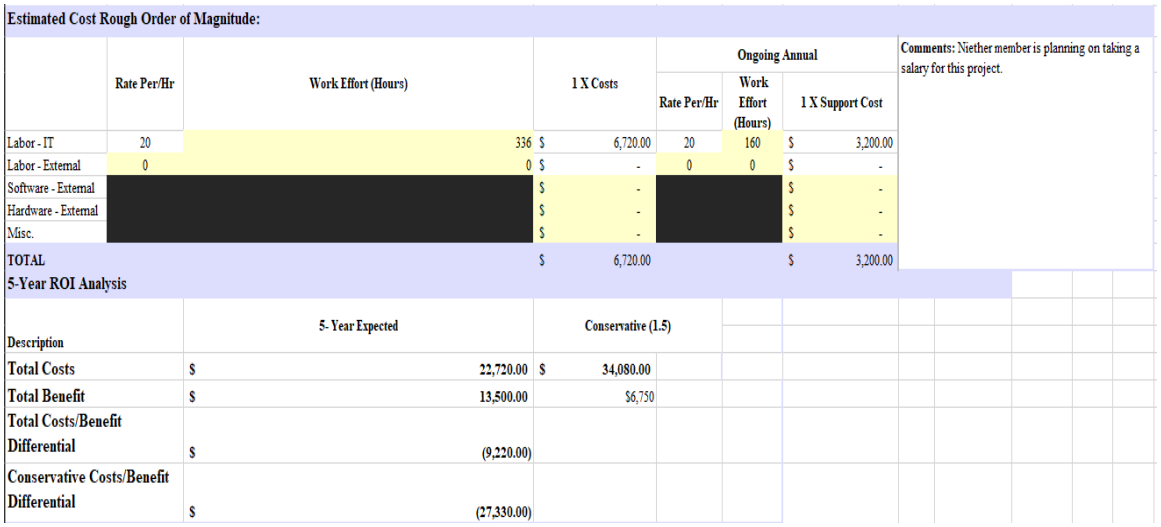
One of the main takeaways that was made from both of us was that even though the test would pass, or it would execute the function that we needed we did not like the way it was done so we went back and altered it. One of the process that we did this with as soon as we had it working was the way that the teacher was viewing the videos that were sent over from the student as we got it working in the way that we originally wanted but it ended up just not working with our format.

We have also been working on the way that we store videos and there has been a lot of debate on if we want to alter it much in the same way that we did in the first example because as it does complete the issue, we need it too, we both believe that it could be done in a better fashion.

PROJECT MANAGEMENT

Budget

This image shows our current working budget and a five-year ROI projection.



Estimated Cost Rough Order of Magnitude:

	Rate Per/Hr	Work Effort (Hours)	1 X Costs	Ongoing Annual Rate Per/Hr	Ongoing Annual Work Effort (Hours)	Ongoing Annual 1 X Support Cost	Comments: Niether member is planning on taking a salary for this project.
Labor - IT	20	336	$ 6,720.00	20	160	$ 3,200.00	
Labor - External	0	0	$ -	0	0	$ -	
Software - External			$ -			$ -	
Hardware - External			$ -			$ -	
Misc.			$ -			$ -	
TOTAL			$ 6,720.00			$ 3,200.00	

5-Year ROI Analysis

Description	5- Year Expected	Conservative (1.5)
Total Costs	$ 22,720.00	$ 34,080.00
Total Benefit	$ 13,500.00	$6,750
Total Costs/Benefit Differential	$ (9,220.00)	
Conservative Costs/Benefit Differential	$ (27,330.00)	

Project Timeline

This table shows the timeline up till now and what we expect to work on in-between semesters.

Task:	Task Name:	Task Duration:	Start Date:	End Date:
Task 1:	Designs/Mockups	21 days	09/02/2020	09/23/2020
Task 2:	Code frontend	9 days	09/24/2020	10/02/2020
Task 3:	Backend mockups	13 days	10/03/2020	10/16/2020
Task 4:	Code backend	33 days	10/17/2020	11/20/2020
Task 5:	Integrate systems	10 days	11/21/2020	12/07/2020
Task 6:	Frontend modification	30 days	12/14/2020	01/13/2021
Task 7:	Backend modification	30 days	12/14/2020	01/13/2021
Task 8:	Design modification	81 days	01/14/2021	04/05/2021
	Oral Presentation		04/05/2021	04/05/2021
Task 9:	Final polish	7 days	04/06/2021	04/12/2021
	Final Report		04/12/2021	04/12/2021
	IT Expo		04/13/2021	04/13/2021

Problems Encountered and Solved

One problem we encounter was our lack of HTML knowledge. This being a website first project, we had to get up to speed quickly. To do this we spent a good amount of time relearning HTML and the components we needed to create our website.

Another problem we encountered was time management. The increasing demands of our classes and work left us with little time to focus directly on the project. To solve this, we worked together to make the most of the time we did have. We worked hard and solved the problems we were facing in the project.

Future Recommendation

- If had more time the main thing that would be looked at was a better tracking/flagging system that could auto alert teachers about set standards like cheating and such. Many proctoring services offer this, and it was investigated, but the type of flagging and alert are a little to jumpy for the standards that we want.
- The flagging feature was one great suggestion that we received from the advisors and maybe a tagging system for students that finished their assignment too fast and could some how indicate this to the teacher.
- If we had to do it all over again, we both agree that we would pick a less complex system as this was a ton of learning new systems and things that we had no experience with. We would keep the same project but format it in a different way.
- We currently do not have any future plans for the project.

Conclusion

One of the main lessons we learned is that coding is much more difficult than previously thought especially when we must learn the language almost completely from scratch. Another lesson learned was time management and the importance of it, as this is a big project and takes a lot of time and effort to be successful. During this project we learned a lot of valuable skills when it came to project management, web development, and website hosting. We gained skills in Firebase, Bootstrap, HTML, and JavaScript. In total this has been a wonderful project for us to learn and grow as IT Professionals.

BACK MATTER

References

"Compliance," February 10, 2021. https://www.proctoru.com/compliance.

"Exclusive Remote Proctoring Features." Accessed April 12, 2021. https://honorlock.com/exclusive/.

"FAQ," February 12, 2021. https://www.proctoru.com/faq.

"HTML Element Reference." Accessed April 12, 2021. https://www.w3schools.com/TAGS/default.ASP.

Khan, Muaz. "Muaz-Khan/RecordRTC." Accessed April 12, 2021. https://github.com/muaz-khan/RecordRTC/tree/master/RecordRTC-to-Nodejs.

"The Leading Proctoring Solution for Online Exams," February 1, 2021. https://www.proctoru.com/.

Mark Otto, Jacob Thornton. "Navbar." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/components/navbar/.

"Online Exam Proctoring with a Human Touch." Accessed April 12, 2021. https://honorlock.com/.

"Online Proctoring and Identity Management Services," February 3, 2021. https://www.proctoru.com/services.

"W3.CSS Reference." Accessed April 12, 2021. https://www.w3schools.com/w3css/w3css_references.asp.

"W3.Css Templates." Accessed April 12, 2021. https://www.w3schools.com/w3css/w3css_templates.asp.

"Buttons." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/components/buttons/.

"Cards." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/components/card/.

"Carousel." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/components/carousel/.

"Close Button." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/components/close-button/.

"Collapse." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/components/collapse/.

"Components." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/customize/components/.

"Forms." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/forms/overview/.

"Introduction." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/getting-started/introduction/.

"Modal." Accessed April 12, 2021. https://getbootstrap.com/docs/5.0/components/modal/.

"Muaz-Khan/RecordRTC." Accessed April 12, 2021. https://github.com/muaz-khan/RecordRTC.