

# **Typerspace: An Interactive Approach to Teaching Keyboard Literacy**

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science  
University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree  
Bachelor of Science, School of Engineering

**Christian Anthony Riewerts**

Fall, 2021.

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Rosanne Vrugtman, Department of Computer Science

# Typerspace

## An Interactive Approach to Teaching Keyboard Literacy

CS4991 Capstone Report, 2021

Christian Riewerts  
Computer Science  
The University of Virginia  
School of Engineering and Applied Science  
Charlottesville, Virginia USA  
[car2xz@virginia.edu](mailto:car2xz@virginia.edu)

### ABSTRACT

In a world where technological development necessitates greater computer skills and the COVID-19 pandemic has forced many schools to transition to online learning, keyboard literacy has become an increasingly important skill to develop from a young age. However, many children fail to understand the importance of this skill, and find it boring and tedious to work on in traditional ways, such as learning the “home position” and practicing by typing sample sentences.

During a 24-hour hackathon, two other college student software engineers and I developed an application called Typerspace which allows users to actively develop typing skills by practicing typing along to the captions of a YouTube video of their choice. The idea behind this was that children, in addition to most other users, would generally be more inclined to regularly use the app to watch their preferred educational or entertainment videos. Additionally, they would be more engaged and more likely to retain the basic keyboard literacy skills from their practice since they would have the ability to choose the subject matter of the YouTube video. We developed the application largely in JavaScript, HTML, and Python, and incorporated other software engineering technologies such as Flask, Heroku, and several APIs which will be discussed in-depth later.

Typerspace largely behaved as desired, but because it has not been publicly released, we have no data to report in terms of its usage or effectiveness in improving keyboard literacy in children. We will conduct and report further analysis following further development and official release.

### 1 INTRODUCTION AND BACKGROUND

Typerspace was developed as a group submission to a 24-hour hackathon. As a result, the problem that the application is trying to alleviate was more so the result of a sudden inspiration rather than a carefully conducted research of issues that we could try to tackle. Despite this, the problem of keyboard literacy was one that all team members agreed was extremely important.

All three of us went to the same school district in Commack, NY, and as a result had very similar experiences in our second-grade computer lab classes. We fondly recalled a typing game which we all were required to play to develop keyboard skills, and reflected on how we recall feeling more engaged and invested due to the entertaining graphics and sounds the game produced. Regrettably, we also agreed that this game would not be an effective way to help adults develop keyboard literacy skills, due to its generally simple and childish nature. This is where the idea of Typerspace was born. By allowing users to combine their unique interests with typing practice, we believed we could serve a much wider group of users from diverse backgrounds and age groups.

As our development team initially theorized, digital literacy skills including keyboard literacy are a very important part of daily life at this point. Workers without these skills are more likely to become victims of structural employment, as they are outmoded due to the changing skillset requirements of the economy. Additionally, adults who are recommended or required to improve their digital literacy have been found to demonstrate hesitancy or outright hostility to programs designed to help them cultivate these skills, possibly due to the insecurity and self-doubt that accompanies unemployment [1]. This emphasizes not only the importance of developing and maintaining digital literacy skills in order to contribute to many working environments, but also the relative importance of learning these skills as early as possible.

The combination of our personal experiences and the digital literacy research we conducted improved the team’s confidence that Typerspace could be an important and useful tool for many people. Allowing the user to choose the topic should encourage young students to practice typing recreationally outside of school, while still making the application approachable for adults who struggle with digital literacy. Ideally, Typerspace will improve users’ keyboarding abilities, which is one of the most important aspects of digital literacy, contributing to a decline in structural unemployment and an overall more efficient economy.

### 2 RELATED WORK

In order to more formally and thoroughly understand the challenges posed by digital and keyboard literacy acquisition, it is helpful to examine current methodologies used to develop these skills. This includes a common typing practice application as well as school policies on teaching and evaluation keyboarding proficiency.

## 2.1 KEYBR.COM

Keybr.com is one of the top typing practice websites to come up for a quick Google search of “typing practice,” and it is immediately apparent why. The application boasts a wealth of useful features which are designed to promote learning, such as statistics for typing speed and correctness, a virtual keyboard which demonstrates the proper “home position” for typing, and the ability to create an account to save your statistics. [2] Additionally, it has a very simple user interface, shown in Figure 1, that allows the user to begin typing right away without choosing and kind of “practice mode” which streamlines the user experience and makes it simpler to interact with.



Figure 1: The main UI of Keybr.com, which the user sees while they practice typing.

One extremely interesting feature of Keybr.com is that by default, users initially start typing with only a small subset of letters, gradually adding more as they progress through different lines of text. This allows users to gradually expand the range of their proficiency when using a keyboard, and also enables the application to collect statistics on specific letters, which can be viewed by hovering over them on the user interface. In fact, clicking the “Profile” button allows the user to see a massive number of statistics about their typing, including daily vs all time stats, change in typing speed/correctness over time, and comparisons to the average user.

The main takeaway from Keybr.com is that it is the gold standard for typing statistic, as it calculates and effectively presents a wealth of data on users’ typing abilities, which may help them to identify weaknesses in their process. The application is not perfect however; it allows users to ignore the backspace key rather than forcing them to delete incorrect letters, which can obviously foster bad habits in learners. Additionally, it suffers from the fundamental problem that Typerspace attempts to overcome, in that it could easily bore users, especially children, and discourage them from regular practice. With no “gimmick” to keep users engaged, it is very possible that knowledge retention may be an issue.

## 2.2 KEYBOARDING CURRICULUMS

Another factor contributing to the suboptimal keyboarding skills of the American economy is the lack of a unified, national policy on keyboarding education. Curriculums are defined on the state level, and these definitions are often imprecise and leave a lot up to interpretation, further contributing to the lack of a decisive keyboarding education plan.

As an example, New York State’s expectations for keyboarding education can be seen in Figure 2. The education plan is very short and imprecise, and the expectations at each grade level are vaguely defined with no success criteria [3]. For example, at grades 5 and 6, it is recommended that “students continue to improve keyboarding skills, with a focus on increasing speed.”

Grade(s)	Keyboarding Instruction Expectations in NYS Next Generation English Language Arts Learning Standards	Page
PreK	Students should begin to explore keyboards.	14
K	Students should explore keyboards.	23
1	Students should continue to explore keyboards.	32
2	Students should be introduced to keyboarding.	40
3 and 4	Students should receive instruction in keyboarding, with a focus on technique over speed.	49, 58
5 and 6	Students should continue to improve keyboarding skills, with a focus on increasing speed as well as accuracy.	67, 75
7, 8, 9/10	Students should continue to improve keyboarding skills to increase speed and accuracy.	83, 91, 101
11/12	Students should demonstrate proficient keyboarding skills.	111

Figure 2: New York State Keyboarding Instruction Expectations

The problem here is that schools cannot evaluate the effectiveness of their program in fulfilling these objectives since they are so vague. A simple change, such as “students should be able to type at 35 words per minute” would provide a concrete expectation for school districts, and empower schools to directly test their students to quantitatively determine if their education plan is meeting the standards. Observing the shortcomings of keyboarding curriculums provides important perspective into how keyboarding improvement is, rightly or wrongly, largely a responsibility of students to work on in their leisure time.

## 3 SYSTEM DESIGN

When considering the decisions made in the system architecture and technology leveraged in the development of Typerspace, readers should remember that development took place fully within a 24-hour timeframe. As such, familiarity and ease of implementation were weighted much more heavily in the decision-making process than they may have been otherwise.

### 3.1 DESIGN AND TECHNOLOGIES

The backend of Typerspace was implemented entirely in Flask, a web framework in Python. This enabled us to create a simple API which would obtain and process the desired YouTube video captions before passing them back to the frontend to be displayed. We made this choice to allow backend processing of the data, and

additionally because members of the team agreed that Flask was a useful framework that we wished to gain more experience with.

Obtaining the captions was a relatively painless process; by using the public YouTube Data API, we were able to get captions and their associated timestamps. From there, processing was quite simple, as the only change that needed to be made was splitting excessively long lines of captions in half to pleasantly fit on the caption carousel, and linearly interpolating the timestamps for the two new “child” captions based on their relative length. At this point, our API could pass the processed captions and embedded video to the frontend to be displayed and interacted with by the user.

The frontend of Typerspace was designed primarily using JavaScript for the functional aspects, and SCSS and HTML for styling and data presentation. I did not directly contribute to the code for the frontend, but the process was relatively simple. The embedded video provided by our API was placed on the screen, and the provided captions were loaded into a simple HTML text carousel in order of timestamp. When the video was played, the current timestamp in the video would be compared to the captions in the carousel to display the appropriate captions at the right time, enabling the user to type along as the video progressed.

### 3.2 CHALLENGES

The main obstacle to the development of Typerspace was, unsurprisingly, the time limit. Twenty-four hours is a ridiculously fast pace to design a fully functional application, and this manifested in certain features not being implemented. We have plans to work on these features in the future, but the fact remains that time constraints were an obstacle to the development cycle.

Another challenge which I personally experienced was a lack of familiarity with certain technologies that we used. Namely, I was only able to meaningfully contribute to the backend of the application, as I had little experience working with frontend technologies. In my opinion, this is a shortcoming of the UVA computer science curriculum. It does an excellent job of teaching students backend languages at the cost of frontend development tools. The only class I have taken at UVA which required any significant frontend knowledge was CS 3240: Advanced Software Development, and even then, the knowledge that I acquired was mostly self-taught since the class didn’t actually teach how to use HTML, CSS, or PHP files directly. I believe that UVA could do its students a service by adding a required course that teaches the basic practices and technologies of frontend development.

Other miscellaneous challenges that we faced were how to handle the characters that could not be typed that were occasionally found in the captions (such as tab and newline) and how to handle multiple languages. In the case of the latter, we chose to limit our scope to English typing practice for the moment, as it was simpler and many YouTube videos did not contain captions in foreign languages.

## 4 RESULTS

Typerspace has not been publicly released yet, so I have no results to report as they relate to user population or effectiveness at improving keyboarding skills. For this reason, a summary of the successfully implemented features of the application, as well its shortcomings is more appropriate.

In terms of successes, the application has a fully functional backend implemented using Flask which, when provided with a valid YouTube video link, is able to obtain the captions of the video and send them to the frontend to be displayed along with the embedded video. The user interface is relatively smooth (see Figures 3 and 4) and allows the user to see their keystrokes represented on the virtual keyboard as they type along with the video.



Figure 3: The landing page of Typerspace, which is where users enter the YouTube video link

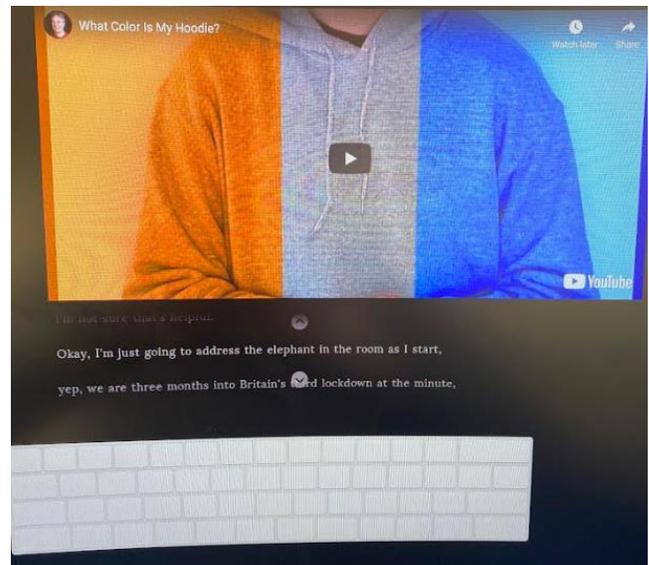


Figure 4: Typerspace interface while user is typing along to a video

Failures of Typerspace are mainly that the caption scrolling makes the transition between lines visually suboptimal, functionality to change text color based on correctness of keystroke has not been added, and the application is not yet hosted online, although this

hasn't been vigorously pursued due to the other failures not being fixed yet.

## 5 CONCLUSION

In brief, digital and keyboard literacy have become massively important in modern society due to both technical advancements as well as the COVID-19 pandemic. For that reason, the acquisition and development of these skills is crucial to almost everybody who plans to work in a non-agricultural setting. Despite this, keyboarding is taught in a relatively informal manner during early education and becomes an increasingly difficult skill to develop as learners age. There is a wealth of online applications to practice typing, but most have the common issue of being too boring to encourage regular use. This results in an inefficient economy where many workers are performing at a suboptimal level, and potentially losing their jobs due to a lack of ability or willingness to improve their digital literacy skills.

I developed Typerspace with a team of two co-developers as part of a 24-hour hackathon to combat this problem. While still under ongoing development, Typerspace will be a fully in-browser web application which allows users to practice typing. To encourage user engagement and regular practice, they choose a YouTube video that they want to watch and practice typing along with the captions as they scroll past. If fully realized, Typerspace has the potential to assist many people of diverse backgrounds and interests develop keyboarding skills when fully released. This will not only improve the quality of life and employability of users, but contribute to a stronger American economy as users become better and more efficient at completing the technical portions of their jobs. We believe keyboarding is a crucial yet undervalued ability in modern society, and believe Typerspace can contribute to a better and more efficient world.

## 6 FUTURE WORK

Development on Typerspace is far from finished. As it currently stands, it is a visually-pleasing skeleton of an application which can display the caption of a given YouTube video, scroll through captions in real time as the video plays, and allow users to type along with them.

There are many more features which the team wishes to implement. Primarily, we would like to include a more robust statistic-tracking system so users can compare themselves to average keyboard users, as well as note their improvement over time. Adjacent to this is adding the functionality to create an account, which will allow the user to sign in from different computers and enable long-term retention of their typing statistics. Finally, some minor tweaks to the user interface will be made, namely changing appearance of the virtual keyboard to better match the site's theme and adding animation to the scrolling captions.

At this point, we will be prepared to find a suitable host for our web application and fully deploy Typerspace, ideally finding some avenue to promote it to potential users so people can benefit from our hard work. When this is completed, Typerspace should increase

the effectiveness of typing proficiency education for both early education as well as senior users.

## ACKNOWLEDGMENTS

I would like to acknowledge the work of Matt Damiata and Benjamin Lapidus, who both contributed heavily to the development of Typerspace. I would also like to thank Professor Vrugtman for her assistance in drafting this paper.

## REFERENCES

- [1] Elizabeth Withers, Gloria Jacobs, Drew Pizzolato, Jill Castek, and Kimberly D. Pendell. 2015. Job Seeking Learners: Digital Literacy Acquisition Case Study. Retrieved October 15, 2021 from [https://pdxscholar.library.pdx.edu/cgi/viewcontent.cgi?article=1017&context=dl\\_research\\_briefs&httpsredir=1&referer=](https://pdxscholar.library.pdx.edu/cgi/viewcontent.cgi?article=1017&context=dl_research_briefs&httpsredir=1&referer=)
- [2] 2021. Keybr.com. Retrieved October 19, 2021 from <https://www.keybr.com/>.
- [3] 2017. The University of the State of New York Guidance: Keyboarding Instruction. Retrieved October 20, 2021 from <http://www.nysed.gov/edtech/guidance-keyboarding-instruction>.