

Lab 1: Inclusive Classroom – Team Gold

Dalton Hanbury

CS 411W

Thomas J. Kennedy

27 September 2021

Lab Version 1

**Table of Contents**

1 Introduction.....	3
2 Inclusive Classroom Product Description.....	4
2.1 Key Product Features and Capabilities .....	4
2.2 Major Components.....	5
2.2.1 MFCD .....	6
2.2.2 Client Application.....	6
2.2.3 API.....	7
3 Identification of Case Study.....	7
3.1 Recordings and Streaming .....	8
3.2 Assignments .....	9
3.3 Future uses of Inclusive Classroom .....	9
4 Product Prototype Design .....	10
4.1 Prototype Architecture (Hardware/Software) .....	12
4.2 Prototype Features and Capabilities.....	12
4.3 Prototype Development Challenges .....	14
5 Glossary .....	16
6 References.....	17

**Listing of Figures**

Figure 1:MFC D.....	6
Figure 2: Sample Live Video Meeting.....	8
Figure 3: Sample Assignment UI Mockup .....	9
Figure 4: Revised Major Functional Component Diagram.....	12
Figure 5: Sample Teacher Site Map.....	13
Figure 6: Student Sample Site map.....	14

**Listing of Tables**

Table 1: Feature comparison of RWP and Prototype .....	11
--	----

## 1 Introduction

Young underprivileged students are adversely affected by having to learn online because of a lack of stable high-speed Internet and because young students lack the knowledge and support at home needed to adapt when the Internet fails. School systems were forced to adapt to new styles of teaching due to the COVID pandemic starting around April of 2020. While colleges have been using resources for online learning for decades, grade schools are not so fortunate. The shift in teaching style is affecting students' ability to learn. Younger students are struggling the most with the shift to virtual or hybrid learning. If the student encounters a technical problem, there is little student can do to fix it. The student will likely ask their parent(s), who may lack the troubleshooting knowledge to fix the problem. The next course of action is to contact the teacher, who, like the parent, likely lacks the skill necessary to deal with a technical issue. The other half of the problem is the lack of stable high-speed internet access at home for lower income families. Roughly 40% of students from low-income homes must complete their homework on public internet connections. These connections tend to have slow connections when many people are on at one time, and frequently there is a time limit as well. A young student relies on their parent(s) or a guardian to take them to public places, which might not always be possible. The solution is a student facing app and a teacher/admin app called Inclusive Classroom. The student view will have a simple UI that is easy to use for children. It will be a native app so that it can run without needing internet access. The student software will have a high level of automation, such as uploading and downloading files, and zipping and unzipping files. These automated features make things as easy for the student as possible. The teacher view will not need to be constrained to a certain device since it is expected that a teacher will likely have good internet access at work. Teachers will be able to log in on any device via login and

password. The teacher interface will be designed with convention over configuration in mind to keep things simple for the student.

## **2 Inclusive Classroom Product Description**

Inclusive Classroom's (IC) primary goal is to increase the accessibility of online learning to low-income students. The solution is a two-pronged approach. The two primary sections will be the student-side flow and the teacher-side flow. The student side will have the ability to passively download and upload, both assignments and lectures. The product will attach a timestamp to completed assignments to enable teachers to determine whether an assignment has been completed in time. The major goal of the teacher-side flow will be to allow the teacher to interact with these students with as little headache as possible. This will be done by automating processes for uploading lectures and sending notifications of live stream status to students. The teacher will also be able to easily review the timestamp associated with the assignment.

### **2.1 Key Product Features and Capabilities**

The Inclusive Classroom (IC) software will have key features and capabilities that distinguish it from the rest of the competition. The biggest concept behind IC is the idea that many students learn in environments without stable internet connections. The client app of IC will be able to operate completely without the internet. If the software is set up by the school or parent(s) with internet access, the student can, in theory, go the entire year without internet until the last day. All teaching materials can be prepared beforehand, and the software will manage all submissions, lectures, etc. to keep the student on pace. If the student can access the internet intermittently throughout the year, the software will automatically synchronize with the backend server, submitting the assignments as if the student had submitted them with internet access the entire time.

At the beginning of the school year, teachers will be able to plan out the entire year, including all lectures, assignments, quizzes, exams, class updates, etc. When the student client software is first set up, it will load all the preplanned data and manage it on the client-side. The student will then be able to follow through with the class in the manner the teacher-designed it. Students will be able to watch the lectures at their own pace, or at the pace deemed by the teacher. Students will be able to complete assignments as they open (weekly, for example), submitting them with or without the internet. As internet is not required through the year, there will be no degradation in student experience. Inclusive Classroom will allow students in areas without internet access, or limited internet access, to learn effectively. If the teacher cannot plan out ahead of time for the whole year, the ability to add in assignments on a weekly basis will help students with limited internet access obtain their needed assignments.

The software will also be available on nearly all operating systems and computers. It will support Chromebooks, as well as the main three operating systems, Windows, Mac OS, and Linux. By supporting these, IC will be an option for nearly all school systems across the United States.

## **2.2 Major Components**

Inclusive Classroom consists of two main components. The client application is utilized by teachers and students. The API handles communication between the client and external databases.

[This space intentionally left blank]

### 2.2.1 MFCD

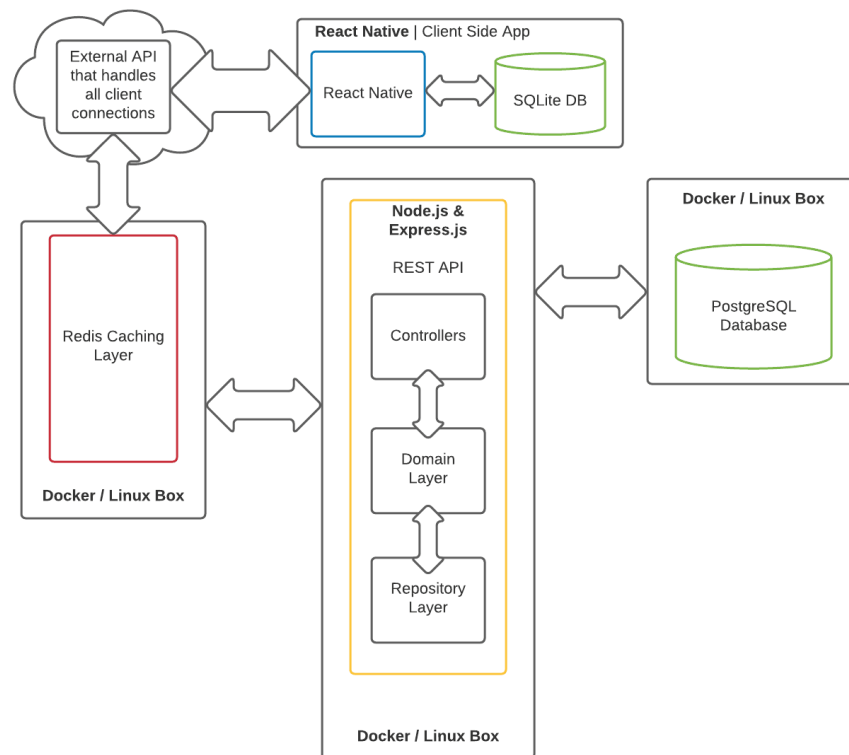


Figure 1:MFCD

The Inclusive Classroom application will utilize the hardware available on the computer.

The API will serve as a handler for all client connections.

### 2.2.2 Client Application

The Client-side application will serve as an interface between the end user and the API as shown in Figure 1. Teachers and Students will utilize the application. A simple and intuitive interface will allow for efficient end-user interaction with the application. The application will be based on a React Native application with an SQLite store. The SQLite store will facilitate viewing of live stream and recordings, recording of live streams, uploading of live streams, downloading/uploading of assignments for Teachers and students, management of classes, and assigning/uploading of grades for assignments.

### **2.2.3 API**

The API will handle all client connections. It will be built with Node JS, Express, Postgres, and Redis. It will handle saving videos to the database, uploading assignments, downloading assignments, getting assignment lists as a priority queue, handle user authentication, and send push notifications. The API will allow for the client application to communicate with the remote database as well as handle user authentication.

### **3 Identification of Case Study**

Inclusive Classroom (IC) is designed with schools and teachers in mind. The product will facilitate online learning for all students, especially those without stable high-speed internet. Stable high-speed internet is defined as internet with less than 1% dropped packets (ICTP). Zoom, one of the major video conferencing software packages, recommends consistent download speeds of at least 3.8 megabits per second to be considered high-speed internet.

[This space intentionally left blank]



### 3.1 Recordings and Streaming

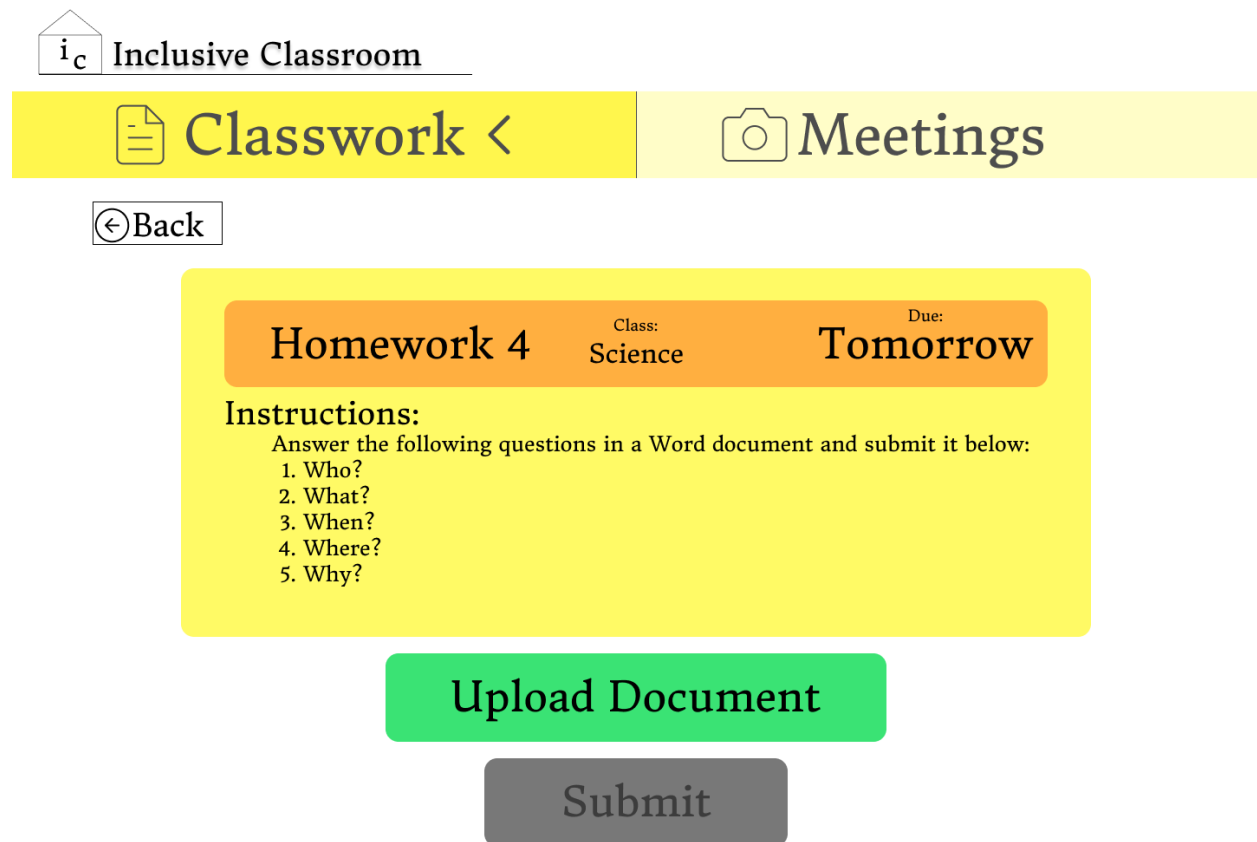


*Figure 2: Sample Live Video Meeting*

Live streaming, viewing, and recording of lectures is a feature that will be simulated during the prototype phase. Integrations will be made with a video meeting API such as Zoom, facilitating live streaming, viewing, and recording of lectures. The video meeting API will be built into the user interface as shown in Figure 2.

[This space intentionally left blank]

### 3.2 Assignments



*Figure 3: Sample Assignment UI Mockup*

The prototype will enable the uploading, downloading, timestamping, grading of assignments, and other necessary features for a simulated classroom environment. Assignments will be handled using the client application's user interface as shown in Figure 3.

### 3.3 Future uses of Inclusive Classroom

With the increasing spread of easily accessible high-speed internet, Inclusive Classroom can be used in the future by families seeking a fallback mechanism in case of faulty internet or loss of service. Inclusive classroom can serve as a reliable alternative or backup for school systems or families participating in online education by utilizing the offline capabilities of the product.

#### **4 Product Prototype Design**

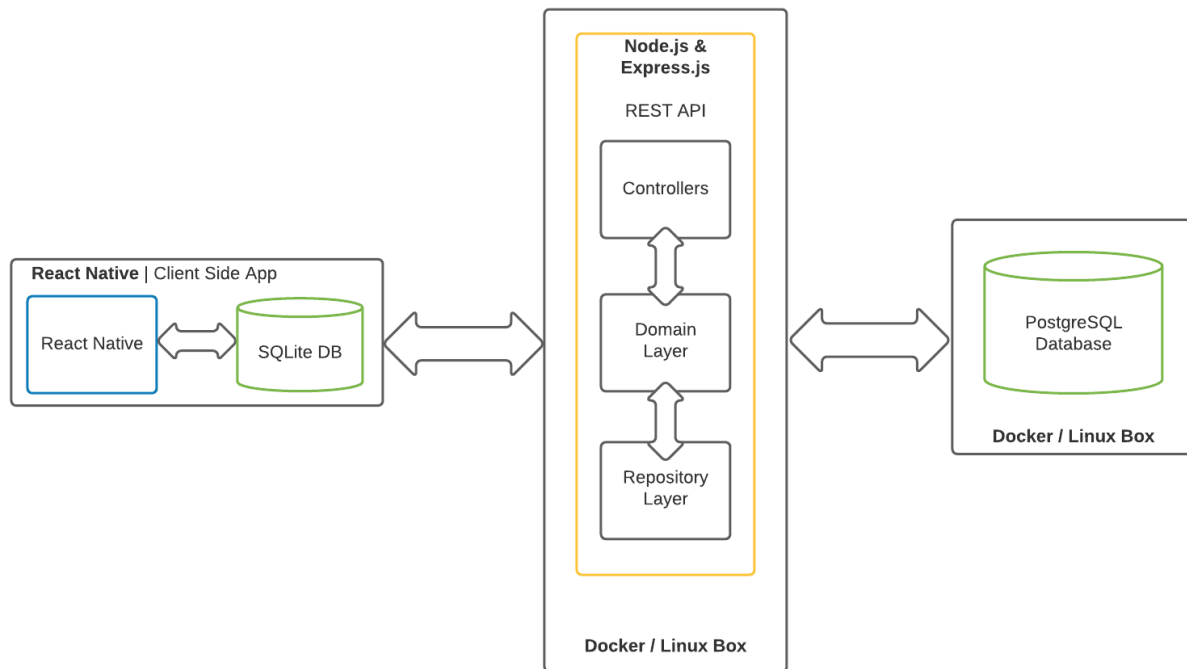
The prototype will include many of the core features of the overall design, focusing on the application and the API. Emphasis will be placed on the automation aspects of the proposed solution with inclusion of a weighted priority queue for assignment downloads, automatic detection of internet, and automatic submission of assignment when internet access is obtained the prototype will implement the following views in order demonstrate the desired automations: student dashboard teacher dashboard, login screen, assignments, and a video conference screen. The technology used in the construction of the prototype will be the same as the real-world product (RWP). Note that many of the features listed in Table 1 for the prototype have limitations due to development and time constraints.

[This space intentionally left blank]

*Table 1: Feature comparison of RWP and Prototype*

Feature	RWP	Prototype
Account Roles	Student, Parent, Teacher, Admin, IT	Student, Teacher, Administrator(limited) only
Automatic Internet Detection	Yes	Yes
Background Workers	Yes	Partial
Complete/Submit Assignments	Yes	Yes
Create Assignments	Yes	Yes, limited in types
Grade Assignments	Yes	Yes
Postdate created content	Yes	No
Recorded Videos	Yes	Yes, may not auto record
Reporting	Yes	No
Timestamp Completed Assignments	Yes	Yes
Video Conferencing	Yes	Yes
Weighted Priority Queue	Yes	Yes

#### 4.1 Prototype Architecture (Hardware/Software)



*Figure 4: Revised Major Functional Component Diagram*

The Inclusive Classroom prototype will share the same hardware requirements as the initial solution, being a computer with Windows 10, Mac, or Chromebook to host the client app. The API will rely on Amazon Web Services' S3, Docker, and Kubernetes. For software, the prototype will continue to rely on the client application and the application will follow the same requirements and usage as listed in Section 2.2.2 Client Application. The prototype API will be built with Node JS, Express, and Postgres, with functionality matching the content listed in Section 2.2.3 API.

#### 4.2 Prototype Features and Capabilities

The prototype will be designed with individualized features for teachers and students each in mind.

## Teacher Site Map

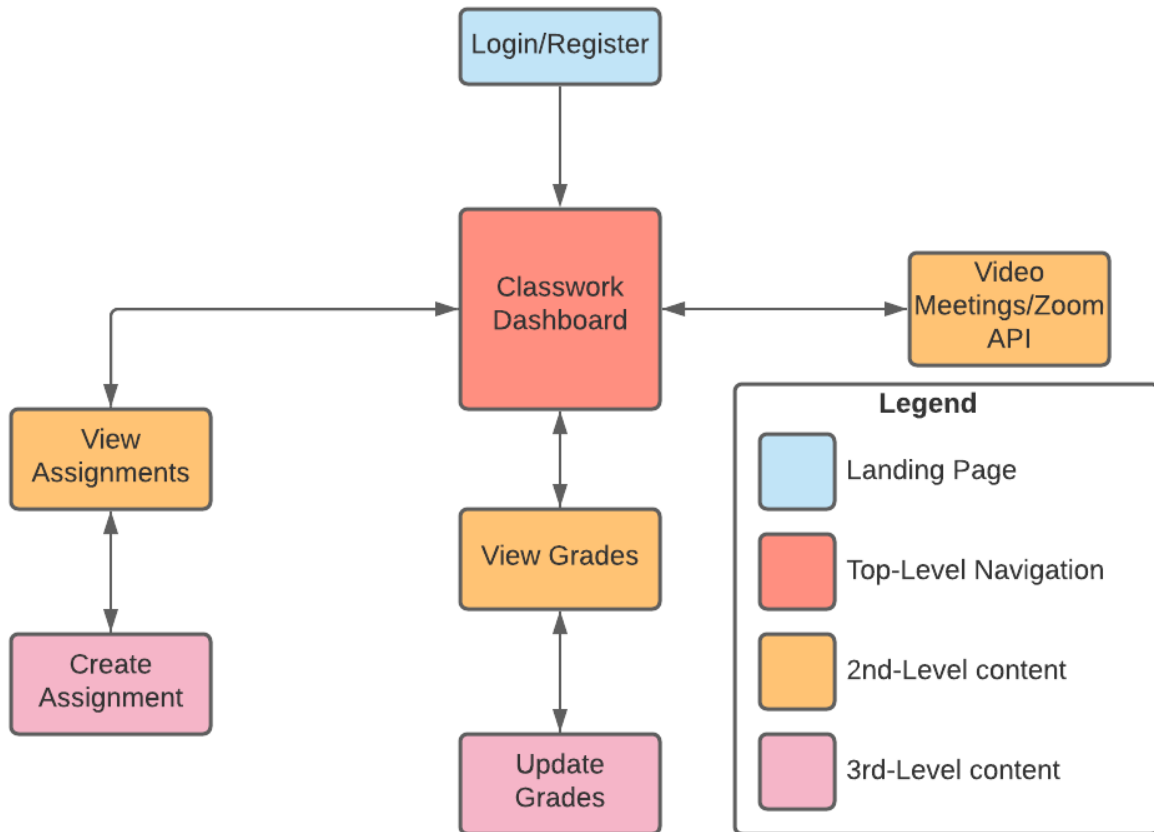


Figure 5: Sample Teacher Site Map

Teachers will be able to create classes, add students to classes, create assignments, schedule zoom sessions, and open zoom sessions for students to join.

[This space intentionally left blank]

## Student Site Map

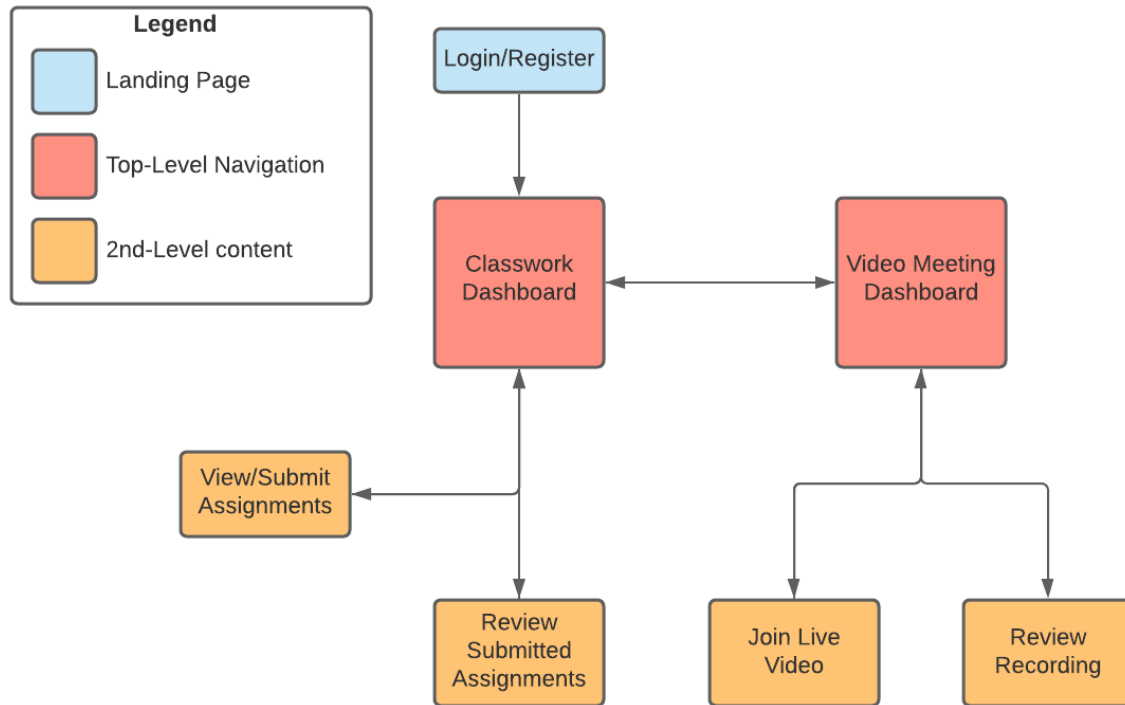


Figure 6: Student Sample Site map

Students will be able to view the classes they are in, view their assignments, make submissions to their assignments before the due date, and watch a lecture live or view the recording afterwards. Teachers and students will share the ability to create accounts and login.

### 4.3 Prototype Development Challenges

The Inclusive Classroom prototype will be limited in scope and features due to several development challenges. The time constraint and length of the course will be one of the most serious limitations, as development will be limited to around 12-14 weeks. Learning of new technologies and frameworks will slow down development due to the unfamiliarity with them and the time required to learn them. Spending time learning and funding to set up CI/CD for deployment with docker containers has the potential to hinder development and testing of the

prototype. Trying to emulate and test the prototype on a school Chromebook setup will be difficult as the Inclusive Classroom team only possesses one Chromebook. Development will be further limited by the need to test the prototype on Mac and Windows in addition to Chromebook. Overall, the development challenges will limit the scope the Inclusive Classroom team and product can cover in the given time allotted.

[This space intentionally left blank]



## 5 Glossary

**High-speed Internet:** Internet with consistent download speeds of at least 3.8 Mbps (Zoom)

**English as a Second Language (ESL)**

**Family Educational Rights and Privacy Act (FERPA):** Federal law that protects the privacy of student education records

**Google Classroom:** “Free web service developed by Google for schools that aims to simplify creating, distributing, and grading assignments” (Google)

**littleLearners:** Former CS 410 group solution that emphasizes simple UI for students in the K-5 age range (Del Razo)

**Stable Internet:** Internet with less than 1% dropped packets (ICTP)

[This space intentionally left blank]

## 6 References

- Anderson, Monica, and Andrew Perrin. “Nearly One-in-Five Teens Can't Always Finish Their Homework Because of the Digital Divide.” Pew Research Center, Pew Research Center, 30 May 2020, [www.pewresearch.org/fact-tank/2018/10/26/nearly-one-in-five-teens-cant-always-finish-their-homework-because-of-the-digital-divide/](http://www.pewresearch.org/fact-tank/2018/10/26/nearly-one-in-five-teens-cant-always-finish-their-homework-because-of-the-digital-divide/).
- “Children’s Online Privacy Protection Rule (‘COPPA’).” Federal Trade Commission, 1 Dec. 2020, [www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/childrens-online-privacy-protection-rule](http://www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/childrens-online-privacy-protection-rule).
- “Chromebook Support.” LCS, [www.lcsedu.net/departments/information-technology/chromebook-support](http://www.lcsedu.net/departments/information-technology/chromebook-support).
- “Classroom FAQ - Classroom Help.” Google, Google, [support.google.com/edu/classroom/answer/6025224?hl=en&ref\\_topic=7175444](https://support.google.com/edu/classroom/answer/6025224?hl=en&ref_topic=7175444).
- Del Razo, Gabriel, et al. “LittleLEARNERS.” Team Orange, [www.cs.odu.edu/~cpi/old/410/orangf20/](http://www.cs.odu.edu/~cpi/old/410/orangf20/).
- ICTP Science Dissemination Unit, ICTP Science Dissemination. “ICTP-SDU Home Page.” ICTP-SDU: about PingER, [web.archive.org/web/20131010010244/sdu.ictp.it/pinger/pinger.html](http://web.archive.org/web/20131010010244/sdu.ictp.it/pinger/pinger.html).
- Kamenetz, Anya, and Eda Uzunlar. “NPR/Ipsos Poll: Nearly One-Third Of Parents May Stick With Remote Learning.” NPR, NPR, 5 Mar. 2021, [www.npr.org/2021/03/05/973373489/npr-ipsos-poll-nearly-one-third-of-parents-may-stick-with-remote-learning](http://www.npr.org/2021/03/05/973373489/npr-ipsos-poll-nearly-one-third-of-parents-may-stick-with-remote-learning).
- Raphael, JR. “Android Apps for Chromebooks: The Essentials.” Computerworld, Computerworld, 19 Feb. 2019, [www.computerworld.com/article/3234533/android-apps-for-chromebooks-the-essentials.html](http://www.computerworld.com/article/3234533/android-apps-for-chromebooks-the-essentials.html).

Section 504 & Students with Disabilities.” Washington Office of Superintendent of Public

Instruction, 2021, [k12.wa.us/policy-funding/equity-and-civil-rights/information-families-civil-rights-washington-schools/section-504-students-disabilities](https://k12.wa.us/policy-funding/equity-and-civil-rights/information-families-civil-rights-washington-schools/section-504-students-disabilities).

“System Requirements for Windows, MacOS, and Linux.” Zoom Help Center,

[support.zoom.us/hc/en-us/articles/201362023-System-Requirements-for-PC-Mac-and-Linux](https://support.zoom.us/hc/en-us/articles/201362023-System-Requirements-for-PC-Mac-and-Linux).

“The 504 Plan.” The Center for Children with Special Needs, 2018, [cshcn.org/childcare-schools-community/the-504-plan](https://cshcn.org/childcare-schools-community/the-504-plan).

VBCPS. “VBCPS Adds 19,000 Chromebooks to Achieve 1:1.” Virginia Beach City Public Schools, [www.vbschools.com/news/archived\\_news/2019/chromebooks](https://www.vbschools.com/news/archived_news/2019/chromebooks).

VDH. “COVID 19 Cases In Virginia.” Virginia Department of Health.,

[www.vdh.virginia.gov/coronavirus/coronavirus/covid-19-in-virginia-cases/](https://www.vdh.virginia.gov/coronavirus/coronavirus/covid-19-in-virginia-cases/). Accessed 20 Feb 2021

Vogels, Emily A. “59% Of U.S. Parents with Lower Incomes Say Their Child May Face Digital Obstacles in Schoolwork.” Pew Research Center, Pew Research Center, 10 Sept. 2020, [www.pewresearch.org/fact-tank/2020/09/10/59-of-u-s-parents-with-lower-incomes-say-their-child-may-face-digital-obstacles-in-schoolwork/](https://www.pewresearch.org/fact-tank/2020/09/10/59-of-u-s-parents-with-lower-incomes-say-their-child-may-face-digital-obstacles-in-schoolwork/).

“Web Applications with Spring.” Spring, [spring.io/web-applications](https://spring.io/web-applications).