# STEM Teaching Application

AJ Johnson
*Dept. of Electrical and Computer Engineering, University of Utah*
*AJ.Johnson@utah.edu*

***Abstract*— The STEM (Science, Technology, Engineering and Math) Teaching Application will provide students with the tools they need to learn different topics in the STEM fields. The application allows users to learn about specific topics they are interested in, or to follow a curriculum for their class. Users can watch short video lessons and tutorials of these topics and then practice what they learned on randomly generated practice-problems. If the application is being used for a class, then the users' practice problem results will be recorded in a database that is accessible by the instructor. All of these tools will provide the students with a responsive learning environment that enables the student to develop their knowledge about various STEM topics. At the core of this application is the commitment to have this technology be accessible to all users, including blind and low-vision users, and those without Internet connections.**

## I. Introduction

In a learning environment dominated by over-sized classes, under-paid instructors, and disengaged students, it's apparent that there is a need for an improvement to the current education model. I believe that teaching technology is the answer to these problems. Specifically, an application that will allow students of all ages to access learning materials that can provide them with the personalized feedback that they are often unable to fully receive from their teachers. Classrooms have long been using technology to aid in the education process (e.g., calculators, projectors, online tests, etc.), rarely is technology used to do the actual teaching. By developing an application that will teach students different STEM concepts, the teachers will be free to spend more time helping students individually, and to provide their classes with a greater variety of engaging activities.

## II. Background

### A. Current Teaching Methods

As expected, there are a wide variety of teaching methods in use today by instructors at all levels of education. Some of the most common teaching methods are:

- Classroom Lectures
- Group Discussions
- Laboratory Projects
- Reading Assignments
- Self-Instruction

Of course, each of these teaching methods has its pros and cons. Classroom lectures are an easy and efficient way to transfer knowledge to a large group of students. Lectures are not as effective as other methods when it comes to material retention, applying the learned content in novel scenarios, and motivation to learn the topic further. Group discussions excel engaging students when the groups are sufficiently small, but face similar problems to lectures when the groups become too large. Laboratory projects can provide students with the opportunity to learn the topics in a hands-on format, promoting retention as well as providing knowledge of how to apply the material in real-world situations. Unfortunately, laboratory projects have the detriment of taking more resources than some of the other teaching methods, such as needed lab supplies, staffing to assist individuals or small groups with the projects, any software and technology needed to process the lab data, etc. These resources can be difficult to maintain, especially for communities that are lacking funding and capital. Reading assignments and online lectures (such as online PowerPoint presentations) have been shown to be as [in]effective as classroom lectures. Finally, self-instruction has been shown to be highly effective, as long as the individual has a structured study curriculum[1].

By using technology to provide a combination of these techniques, students will be able to reap the benefits of these teaching methods while bypassing the methods' shortcomings.

### B. Existing Teaching Technologies

Interestingly, most "teaching technologies" that are readily available on the market today are not, in fact, technologies that teach, but simply technologies that assist teachers in areas such as test distribution and proctoring, assignment collection, schedulers and more. These ill-named teaching technologies do not perform any of the teaching, however, so there is a large portion of the education industry that technology has not yet flourished - namely, the teaching!

Some websites such as Khan Academy (khanacademy.org) and Youtube (Youtube.com) and some academic institutions including MIT [2] and the University of Utah have begun to record audio and video lectures, and are publishing them online to be used by the general public. The topics that these free lectures cover is vast, yet as stated before, self-teaching methods such as viewing online lectures requires highly structured study in order to be effective. So while these resources are freely available to the public, they do not provide the necessary structure in order to be highly effective teaching tools.

Studies have shown that external feedback can have a significant impact on learning, depending on the type and quality of feedback [3]. Unfortunately, not all students respond well to the same type of feedback. If two students -

Student A and Student B - are each scolded for performing poorly on a class assignment, they will likely take steps to avoid being scolded again. Student A may study harder in order to receive a better grade on the next assignment, while Student B may attempt to cheat in order to receive a higher grade. Both students will avoid a scolding on the next assignment, but it is clear that they responded very differently to the feedback that they received. In the case of Student B, the feedback clearly had an undesirable effect. As Dylan William phrases it, "You would have been better off shutting up than actually giving the feedback." [4]

It is critical that the feedback that is provided by a quality teaching technology be able to adapt to individual needs. For example, in an application that has students answer simple math questions and provides them with a message of whether their answer was correct or incorrect, some students may respond better to a simple message them telling them they were wrong and providing them with a new problem [Fig. 1], while others may respond better if it was pointed out where they may have gone wrong in their answer, and encourages them to follow along with an example problem [Fig. 2].

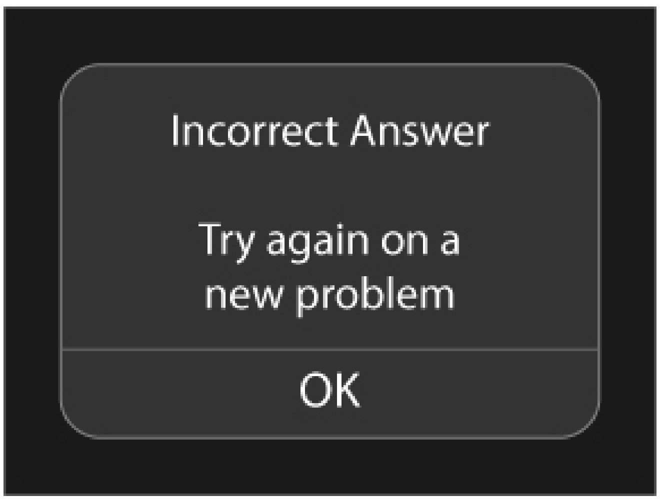


Fig. 1: Sample message dialogue for an incorrect answer that will provide the user with a new problem.

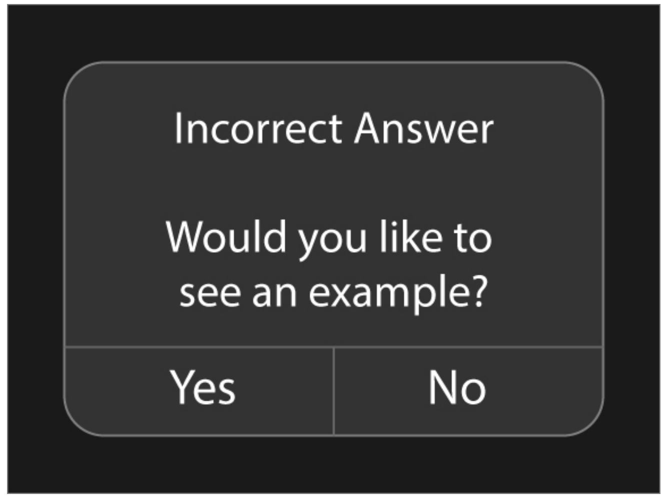


Fig. 2: Sample message dialogue for an incorrect answer that will encourage the user to follow along with an example problem.

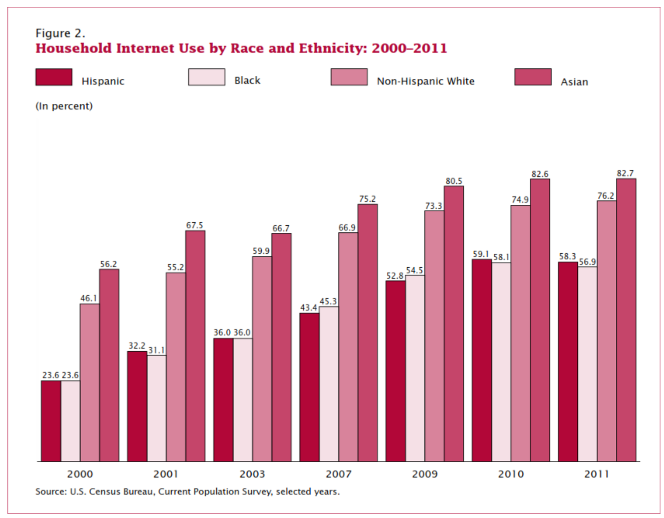


Fig. 3: 2011 Census Data displaying Internet access across racial lines

Interactive, teaching technologies, such as iPad applications, provide students with instant feedback, but are often quite limited in the topics that they can teach. That said, studies have shown significant increases in academic performance measures, including literacy, mathematical ability, and graduation rates, simply by incorporating iPads into the learning curriculum [5]. As with the laboratory projects stated before, using iPad technology can prohibitively expensive. For example, to supply an average Utah elementary school class [6] of 24 students with new iPad Airs, it would cost $12,000 - roughly 25% of the mean of Utah's elementary school teachers' salaries [7]. While iPads and other tablet technologies may effectively enhance education in school districts that can afford them, the cost makes this inaccessible to low-income communities.

Another large barrier to incorporating technology further into students' education is the lack of an Internet connection in many households. In 2011, only 71.7 percent of households reported accessing the Internet [8]. It is unrealistic to assume that every student will have access to an Internet connection at home. Access to an Internet connection is also strongly correlated to race [Fig. 3], so by using technologies that require an Internet connection, these are further disadvantaging groups that are already under-served in today's society. In order to have a teaching technology be accessible to additional populations, it needs to be able to function for students in homes without an Internet connection.

Students with visual impairments and disabilities are another largely marginalized group when it comes to teaching technologies. Many of the technologies on the market today are less than ideal for non-sighted users. Whether the applications do not provide necessary accommodations (i.e., descriptive audio) or the technology itself is inaccessible, such as touchscreens (though, some exciting research is being done to make touchscreens more accessible to blind and low-vision users citeVanderheiden:Blind), the current technologies are rarely robust enough to fully serve users with little to no vision.

### C. Motivation

As discussed above, it is apparent that teaching technologies could be greatly improved and further incorporated into the education process. Specifically, a quality teaching technology will need to be able to:

- Teach a student a specific topic
- Allow the student to practice/apply the new knowledge
- Provide feedback to the student
- Adapt the type of feedback based on the student's needs and performance
- Function sufficiently without an Internet connection
- Accommodate blind or low-vision users natively
- Operate using free or low-cost technologies
- Allow the instructor to view the progress of the students

## III. Application Outline

This teaching application is a very broad project which will incorporate many complex components, including: an expansive database, design and integration of online and offline student clients, intelligent/learning feedback handlers, robust and intuitive GUIs, flexible lesson schedulers, and native accessibility features for blind users and users with disabilities.

### A. Application Architecture

The application will be divided into five main components:

- Web Server
- Database
- Online Student Client
- Offline Student Client [Installer]
- Instructor Client

*1) Web Server:* The web server will likely be implemented on a virtual machine, running Apache web server. The server back-end will be constructed using the Rails framework, since Rails is exceptional at rapidly prototyping web project, it should be a great fit for this project. This application is anticipated to change and evolve significantly over the course of its construction, so Rails will provide the necessary adaptability that is needed to keep abreast of the changes.

*2) Database:* The database component of this application will be exceptionally wide in scope. Currently, it will contain several tables in order to track the necessary data:

- *Instructors* — instructors' information (name, email, id, etc.)
- *Students* — students' information (name, email, id, etc.)
- *Courses* — course information (course name, description, institution, instructor, current assignment, etc.)
- *Enrollments* — enrollment information, specifically pairing a student with a course
- *Assignments* — assignment information will be contained here, which will likely be in JSON, or some other easily-parseable format
- *Student_Assignments* — assignments that have been completed by users, number of problems completed, accuracy, grade received, etc.

This database will likely be implemented using sqlite3, as that is a technology that is easily used with a Rails web server. In the case that Rails is not used for the web server, MariaDB or a MySQL database will most likely be used.

*3) Online Student Client:* The online portion of the student client will enable the students to follow lesson plans, watch the instructional videos, do practice problems, complete and hand in assignments, and possibly have a live discussion (text or video) with the instructor. These are the proposed goals for the online client, and features may be added or removed as the program evolves.

The online student client will simply be accessing webpages that will be built with real-time, interactive webtools including AngularJS and will be responsive and will adapt to the users' behavior with AJAX. The problem generating, learning algorithms and other areas of the program that are resource-intensive in terms of computing power will be performed client-side, while the server will only be responsible for updating the database and fetching files and information for the client.

*4) Offline Student Client [Installer]:* In order to allow students without an Internet connection at home to be able to use this technology to its fullest extent, they will be able to download their daily assignments onto their laptop at school (any institution using this teaching technology will be required to have an Internet connection that is accessible to students). This does also mean that students will need to have access to a laptop or similar device that they can take to and from their institution daily. Loaner-laptops could be used and are not nearly as cost-prohibitive as iPads. A bare-bones Samsung Chromebook costs only $159.97 and would certainly provide enough computing power for this simple learning application.

Generally speaking, offline assignments will reflect the online content associated with the current assignment provided by the instructor. Students will use their laptop to download the daily assignment(s), take the laptop home to work on the assignment, and the next day the results of the assignment (and the students' learning patterns/behaviors) will be uploaded to the database.

The offline client would download the assigned lesson for the day, providing the student with the needed instructional videos, practice problem framework, and other details that were necessary for the current lesson/assignment. Unfortunately, this does mean that a significantly more finite amount of information is readily available to students who will be working offline, as they will be unable to advance to other topics, review past topics, or view anything other than the current working material, depending on how the application is designed. Ideally, it would be light-weight enough that the past several assignments as well as the next several assignments would be accessible from the offline version as well, but that would increase efficiency when downloading assignments.

This offline component could be one of the most challenging portions of the client to implement. The offline application would likely be written in C# or Java, as those both provide many tools to process downloads and incorporate them. The downloads will consist of video files, text files, possibly static HTML pages (or the offline application could simply be run with a web-browser, using only local files!) As the design progresses, there will be clearer benefits and detriments to the available options.

*5) Instructor Client:* The instructor's client will be very distinct from the two before. The instructor will most likely simply use an Internet browser to access the webpages and utilize a GUI to perform their tasks. Most of their tasks will be along the lines of creating lessons, distributing and collecting assignments, and possibly providing live, teaching support to the students via the online client. Most likely, the instructor's client will follow utilize the same backend as the students' client, because there will be a significant amount of resources that are shared between the two that would be inefficient to port into a different language/environment. Specifically, using AngularJS and AJAX calls to smoothly interact with the pages would be ideal.

### B. Cost

Aside from the virtual machine for the web-server, there should be very little overhead cost for this project. The main cost of this project will be in the developers' time. There is a significant amount of design work that needs to be done before this application can be implemented. Perhaps a designer may be incorporated into the project in order to make it as intuitive and attractive as is realistic. Consulting behavioral psychologists could also be incredibly beneficial when developing the learning algorithms that provide feedback to the students. There is potentially a very large knowledge gap in the topics that this application may be teaching at some point and the knowledge of the designers, but at the onset of this project, the only topics covered will be kept at a high school level or below.

### C. Schedule

The timeline of this program is relatively fluid, but should follow these rough guidelines:

1) The first 10% of the timeline will be devoted to guaranteeing that the design specifications are thorough, well-constructed and complete.
2) The next 5% of the timeline will be implementing the architecture for the project. Acquiring and setting up the needed virtual machines, establishing documentation and standards to be used for the project, starting a versioning system for the application and so forth.
3) The next 15% of the timeline will focus on constructing a VERY basic, walking skeleton of the online components of the project as well as a simple offline client. Creating a bare-bones GUI to retrieve a video file, generate very basic problems, and report the results to the database.
4) At this point, 35% of the allotted time will have passed, and having a walking skeleton will allow us to very rapidly begin designing components for it. Creating several sample lessons, some problem generators, improving the GUI, creating the instructor's interface to build classes and lesson plans, etc. This rapid design period will last for approximately 35% of the remaining time.
5) The next 15% of the time will be used for refactoring. After having gotten so far into the code, there is a large likelihood that some large portions of the design would be able to benefit from rethinking and redesigning these in order to make them more efficient, usable and maintainable.
6) The final 10% of the timeline will be used to implement any stretch features that seem like they would help make the application more "complete."

### D. Risk

With the group size for this project still unfinalized (estimated to be 2-5 individuals), the risk associated with this project is similarly unknown. There are certain aspects of the project that will currently take a significant amount of time to implement that could likely be bypassed in the final version. Specifically, the learning algorithms to be used for student feedback could be replaced with a simple, standard feedback method ("correct!" or "incorrect, try again.") These would certainly be less helpful to the users, but would provide enough for the user to get by. Similarly, the offline or online components could be merged together into a single offline application. The purpose of this project is to provide quality, accessible, educational technology to as many populations as possible, so it is imperative that the offline portion of the project be created. Without the offline component, this project once again reinforces the current trend of upper-income students receiving a better education than lower-income students.

## IV. Project Implementation and Design Details

### A. Web Server

The web server(s) will have two primary roles in this application:

1) The web server must allow new users new learn about the application; get a summary of the product, see examples, and ultimately create an account if they would like to use the application.
2) Existing users must be able to perform the necessary tasks by using the server while logged in. For users who are instructors, this means that they must be able to to organize classes, schedule assignments and plan coursework. For users who are students, they must be able to access their courses' assignments, download any resources they need to work at home and/or offline, and engage with the class in other ways (e.g., uploading assignments, participating in online discussions, etc.).

The web server will also be integrated into the database, but that will be discussed further in the Online Student Application section.

To actualize this web server, I will likely use a cloud computing server such as Microsoft Azure, since I have familiarity with their services. Additionally, I currently am paying for web-hosting from the company Siteground, which hosts my logs for this project (ajjohnson.lgbt/project-logs), so I may be able to get free hosting from them if I approached them and informed them that I would be doing this for my project. The server would likely be an Apache web server, and may utilize the Ruby on Rails framework as a backend for the content. The Rails framework is flexible enough to manipulate it to perform the needed functionality, but abstracts away a large portion of the tedious tasks (e.g., database accesses, creating paths, etc.)

Finally, the web server would need to be created with the flexibility needed to be scalable in the future. A system such as this would need to be implemented slightly differently in order to evolve to a distributed application on multiple servers, but during the course of this project, the demands of a single course or two (roughly 50 students) shouldn't overload a single server, so while it will be developed with scalability in mind, it won't necessarily be implemented in this iteration of the project.

### B. Database Backend

As of right now, the database that this project will use will likely be Sqlite3, which is highly compatible with the Ruby on Rails framework that was discussed in the web server section. As all databases, this will be organized into several tables as follows:

- User
  - *user_id, (Primary Key, auto-incrementing, integer, not-null)* — This unique identification will be automatically assigned to users as they create their accounts.
  - *username, (Unique, varchar, not-null)* — This unique identifier will be used by individual users to login to the application.
  - *hashed_password, (Varchar, not-null)*
  - *email_address, (Varchar, optional)* — The email address is optional, because some of the users of the application may be fairly young and may not have an email address yet. Additionally, many people just don't have email addresses, and since this application doesn't provide them with an email address, that could just be another barrier to the accessibility of this application, if one was required.
- Profile Information
  - *user_id, (Primary Key/Foreign Key)* — Associated with the user_
  - *first_name, (Varchar)* — First name, for profile information.
  - *last_name, (Varchar)* — Last name, for profile information
  - *public_name, (Varchar)* — This name will be the name that the public will be able to see when they are interacting with this user. For whatever reason, some users may go by a different name than their given first and last name, so they will be free to enter it here. This is one of the places where I can incorporate social justice into the application, and make an application that a greater amount of users can enjoy fully.
  - Any additional profile information that could be useful. Contact information, address, etc. This will likely change over development as the project comes into its own, and we discover what columns are more necessary for the application.
- Course
  - *course_id, (Primary Key, auto-incrementing, integer, not-null)* — This unique identification will be automatically assigned to courses as they are created.
  - *course_name, (Varchar)* — The specific name of the course (i.e., CS 3992)
  - *institution, (Varchar or int..., optional)* — Institutions may be attached to their specific courses, depending on how the program evolves. The instructors of a course shouldn't necessarily need to be professional instructors attached to an institution, so this field isn't required, but it could help in the future if we find that institutions want to be linked to their classes.
  - *start_date & end_date* — If these dates for the course are known, then they can be posted here.
- Enrollment — Used to associate users with specific courses. Will have a "role" column that allows users to be divided into instructors and students (and potential further divisions from there).
- Assignment Specifications — Used to define assignments and associate them with courses. Assignments will have a specific, unique identification that will further be used to associate these specifications with students submitting their assignments.
- Assignment Submissions — Contains the instances of assignments that students completed and submitted to the application. The exact definition of this table will be quite vague until the application is further developed.
- Learning Feedback — This table will be used to specify how well/poorly students responded to the different types of feedback while working on their assignment. For example, it may contain several columns of different types of feedback (e.g., praise, disappointment, challenges, helpful hints, etc.) and have an integer representing how well the student responds to each type, with a higher value signifying a more positive response. Once again, this table will be very nebulous until the application reaches a further state in development, and after a behavioral psychologist is potentially consulted.

## C. *Online Student Application*

Upon further consideration of the overall goals of this teaching application, the online component has move from being the primary focus for users to being a secondary component to be worked on once the Offline Student Application (described in the next section) is functioning. As stated before in this proposal, a main focus of this assignment is to create an application that is as accessible as possible for users, and treating the offline portion as an extra *feature* could perpetuate the issues that people without Internet connections are already facing.

The online application will be created using webpages made AngularJS, as it handles CRUD (Create, Read, Update, Destroy) applications very well, and this application should follow that pattern. AngularJS allows for many abstractions to take place so that the developer can spend less time on the trivial details (i.e., AJAX requests, associating views and models, etc.) and spend more time focusing on creating the application.

The application will allow students to watch videos, using HTML5 video, complete assignments (which will just be dynamic HTML forms), and receive various kinds of feedback. AJAX requests will be submitting the data as the assignment is completed, and a live chat feature may be implemented as well, so that instructors can have a one-on-one chat with online users. However, this does bring up the discrepancy again of who this application will be benefiting most, and extra help for the online students — despite how simple it may be to do — would not achieve the goal of providing all students with the same opportunities to learn. Having money (and an associated Internet connection) should not be the reason that someone receives a better education than someone else.

## D. *Offline Student Application [Installer]*

The offline student application will be where the bulk of the work takes place in this project. It will have three main components associated with it:

- An offline, desktop application — This application will be written in Java or C#, most likely, and will allow students to perform all of the same actions that they could online, assuming that the associated lesson files are downloaded onto their system. They will be able to view instructional videos and examples, complete assignments and receive the various kinds of feedback in the application, tailored to their learning styles.
- An online data handler — This data handler will be a part of the web server that allows the client's desktop application to submit completed assignments (specifically assignments that had been completed offline), and will download the next lesson(s) and assignment(s) that are needed for the course. This downloader may just be as simple as an HTTP GET request from the client that retrieves the associated lesson and assignment files, but it could be a full-on installer that downloads .jar files or .dll files, for Java and C#, respectively.
- Downloadable lesson files — These files are the data that defines what lesson and assignments are happening in the class currently, and includes the associated video lessons, specifications for assignments (perhaps in JSON), and finally the problem generator(s) needed for the assignment. Problem generators will be assignment specific, and will likely be some sub-classed object with a method GenerateProblem(), which is specified in this lesson file. The problems that are provided to each student will have a degree of randomness on them, to discourage sharing answers between students, and encourages actually learning the problem oneself — or at least coming up with a more clever way to cheat!

## E. *Instructor Application*

This will be a component of the web server, but it is separated here because it is going to be a very time-consuming portion of the application. Essentially, this portion of the application will allow instructors to control the lesson plans and assignments that their classes will be following, and it will also allow instructors to view assignment submissions, see how individual students are performing, understand their students' learning style, and more.

The application essentially be an administrator's view of the course. There will be form tools in place in the web pages to allow the instructor to update the lesson plans at the press of a button. They can design their own new lessons, or pick from lessons that are in the online collection (created by other instructors, or by the administrators of the website, AKA: me.)

## References


[1] J. Olsen, R. Saracci, and D. Trichopoulos, Eds., *Teaching Epidemiology: A Guide for Teachers in Epidemiology, Public Health and Clinical Medicine*, 3rd ed., 2010.

[2] MIT. (2015) MIT OpenCourse Ware. [Online]. Available: http://ocw.mit.edu/courses/audio-video-courses/

[3] S. Narciss. (2006) Feedback strategies for interactive learning tasks. Narciss__S.__2008__Feedback_strategies_for_interactive_learning_tasks.pdf. [Online]. Available: http://www.phil.uni-passau.de/fileadmin/dokumente/lehrstuehle/narciss/

[4] M. Scherer, "Perspectives/Finessing Feedback," *Feedback for Learning*, vol. 70, no. 1, pp. 7–7, September 2012.

[5] Apple. (2014, April) iPad in Education Results. iPad_in_Education_Results.pdf. [Online]. Available: https://www.apple.com/education/docs/

[6] L. Schencker. (2011, February) Analysis: Smaller class sizes, better results. class-schools-sizes-students.html.csp. [Online]. Available: http://www.sltrib.com/sltrib/home/51193425-76/

[7] R. Winters. (2011, September) New report says utah teacher salaries may not be so bad. utah-teachers-pay-report.html.csp. [Online]. Available: http://www.sltrib.com/sltrib/news/52502415-78/

[8] T. File. (2013, May) Computer and internet use in the united states. p20-569.pdf. [Online]. Available: https://www.census.gov/prod/2013pubs/