**DIGICLIPS CAPSTONE PROJECTS**

7/27/2022

Dear Students & Professors,

Thanks for your interest in the student participation projects that DigiClips offers. This is our 7th year of sponsoring projects. If you choose to work on one of our listed projects or to create your own, you will be getting experience working with the entrepreneurial software startup portion of our company. This is a different experience than that provided by a large corporate company. In a lot of ways it is the old "Small Fish in a Big Pond or Big Fish in a Small Pond". The typical corporate software careers I have seen student coders enter into involves being assigned a small area of coding for a few years and then being bumped up to administration of less senior coders or even other administrators. This can be pretty financially rewarding & gradually lead to more influence of what the company is doing, all depending on your ability to advance in the company bureaucracy. Getting advice and/or assistance in a large company can be chancy. In a startup, on the other hand, you are typically involved in making decisions about what the company is doing & where it is going as well as coding right from the start. Also, being a smaller company, you can directly deal with Bob Shapiro (CEO) & Henry Bremers (Senior Software Engineer) when you need help with design or implementation. We are available 24/7 by phone, SMS or email. An article on Linkedin recommends that you try both during your career: <https://www.linkedin.com/pulse/small-fish-big-pond-gaurav-kumar/>

DigiClips is a media monitoring company. We monitor TV, Radio, Magazines, Newspapers, Social Media, Blogs, Web Media etc. 24/7 in 210 USA DMA's (Designated Market Area) and internationally. Clients give DigiClips keywords or other information about what they want monitored and then DigiClips sends them email research reports with URL links, clips, metadata, Nielson ratings, etc. whenever search criteria are met. Here is the summary of our current media monitoring structure:

1. **BACK END RECORDING MACHINES**
	1. The **Television recording** backend uses ffmpeg and several computers to record broadcast television stations 24/7. Recordings remain on these machines, but a copy of closed caption & other text such as video-to-text are sent to the MySQL Server. We are also working on cable and other sources. These machines stay in the markets they serve. Our software and ffmpeg API are written in C.
	2. The **Radio recording** back end uses ffmpeg to record streaming radio stations 24/7. Recordings remain on the machine, but a copy of the speech-to-text is sent to the database server. Speech-to-text is currently being moved to a more modern API & needs a fair amount more work. It is written in C.
2. **FRONT END WEB SERVERS**
	1. We have a basic **Informational Website** that can generate customer request emails to us. It uses Angular 12, Typescript, Javascript, Express, Node.js and React. It will hopefully soon be moved to AWS.
	2. Under development, but mostly functional, is a **Search Engine Website** for our customers to use to create their own media searches. We or our customers can use this website to do searches or set up email alerts to send us clips, etc. when a search criteria is met. This code can also do web scraping to retrieve newspapers, magazines, blogs, etc. It uses Github, Ffmpeg, Angular 12, Typescript, Javascript, Express, Node.js and React.
	3. Under development & mostly functional is an **Administration Website** that displays errors & other statistics about how the recording machines are running. It also keeps track of users & changes configuration files on the recording machines. It uses Angular 12, Typescript, Javascript, Express, Node.js and React.
3. **A** **MySQL DATABASE SERVER** is used to accumulate and search text from the recording machines for use of the Front End Websites. It also receives error messages from the recording machines which are then used by the Administration Website.

Our services are used by:

* Public relations firms, company media relations departments & others that do press releases to the press or when certain issues come up like a media crisis they need reports about and clips of their segments, hit reports and media analytics. Note: To create the media reports and to get some Television Video and Radio Audio segments that we aren’t recording, we work with some of our competitors.
* Lawyers to keep track of what the media says about their clients
* Advertising agencies that need to know when their clients' ads are aired and their clients' competitors' ads. Plus they need proof of performance of their own clients advertising time purchased.
* Politicians that need to know what the media & opponents are saying about them
* School districts Health care companies Hospitals Universities
* Government agencies local - city - state - federal - Foreign Countries Embassies
* Competitors that need our high def recordings (at a wholesale price)
* General Public

**Projects**

Projects can consist of one or more of the numbered items. Some items are easy, pick several of them. Some are hard mult-semester projects that will need future developers to finish. That’s okay too. You can also even design your own projects that we haven’t even thought of yet. Our basic goals are to make defining a search more intuitive, gather as much data as we can, including that which is mostly missed by the current Media Monitoring Industry and to generate media results reports that are user friendly.

**SEARCH ENGINE FRONTEND:** Searches are presently done with our Search Engine and through an arrangement we have with one of our competitors. Keyword email alerts are emailed to us with reports that we can then email directly to our clients. The competitor also provides Radio and TV segments that are not recorded by us in our home market. Clients can then go through the reports we send them and ask us for edited copies of the segments from Television and Radio. Currently, much of this monitoring is done through closed caption text, but DigiClips is automating these tasks and adding new services that other media monitoring services don't provide. Closed caption text misses a fair amount of data. We are working on adding television speech-to-text, video text recognition, image recognition and lipread-to-text to capture data that is currently being missed in the monitoring industry.

The website can search the DigiClips DB for text from the Radio and TV recording machines so it can link to these machines to retrieve recordings and stream the Television or Radio segments. This code can also do web scraping to retrieve newspapers, magazines, social media, blogs, web media, etc.. and then put this data into the DB. Users can view their results directly on the website or in the email alerts they receive. It is written using GitHub, Ffmpeg, Angular 12, Typescript, Javascript, Express, Node.js and React. Some standards we use are MP3, MP4, WAV, ASS, SRT, VTT. Some tools we use are OpenVPN, SSH, AWS Lightsail and Chrome Remote Desktop. This website currently runs on a computer on our network and in the cloud (AWS Lightsail) [http://3.13.169.109:4200](http://3.13.169.109:4200/). Note: Most of the following links require you to be a logged in user to see their content.

Current search menu structure for search page:

**Language Translation:** [**http://3.13.169.109:4200/languagetranslation**](http://3.13.169.109:4200/languagetranslation)

NOTE: Not sure where this should go needed language translation for everything

**Log in**: <http://3.13.169.109:4200/login>

NOTE: Go to submenu and create new users that have to be approved

**Search Page:** [**http://3.13.169.109:4200/search**](http://3.13.169.109:4200/search)

NOTE: This click goes on the Search Page

**Option Page**: <http://3.13.169.109:4200/optionpage>

NOTE: This click goes on the Option Page

**Email Alerts:** [**http://3.13.169.109:4200/emailalerts**](http://3.13.169.109:4200/emailalerts)

NOTE: This click goes to the Email Alert Page:

**User Profile:** [**http://3.13.169.109:4200/usersprofile**](http://3.13.169.109:4200/usersprofiles)

NOTE: This shows the user their profile as there are on other webpages.

**Help** [**http://3.13.169.109:4200/help**](http://3.13.169.109:4200/help)

NOTE: This has been created and was working. Needs to add on more as we work through this project to help the users.

**Contact Us:** [**http://3.13.169.109:4200/contactus**](http://3.13.169.109:4200/contactus)

NOTE: To send and contact the Administrators with problems and comments.

**Log Off:** <http://3.13.169.109:4200/logoff>

NOTE: User clicks on this and can log off the site.
**Media Analytics:** <http://3.13.169.109:4200/mediaanalytics>

NOTE: The SE is on GITHUB and working on digi-frontend & AWS. It needs a little more work. Rough draft on: <http://www.digiclipsinc.com/digiview/FilterBoxReport.htm>

1. It is written in Angular 12 & needs to be updated to Angular 13. All node packages will also need updating.
2. Add video-to-text & audio-to-text as in addition to the current closed captions to search.
3. Make reports more user friendly, possibly with graphic representations of some data.
4. Artificial Intelligence to assist user design of queries.
5. Use backup captured data to replace data missed DNR (Did Not Record).
6. Improve query options page to further refine the selection of just the data needed. For instance some TV searches don’t need to display the video, they just need to show the time it ran & audience numbers. (There are already a lot of available options).
7. Improve email alerts setup page and do a lot more field testing to make sure it is working properly, delivering correct links, clips, etc.
8. Keep track of and show Network affiliates on reports.
9. For our advertising clients check on & download their competitors commercials. Add media analytics showing when, where, costs, viewer ratings, etc. of these competitors’ ads. Could use graphics to display this info. Commercials Proof of performance of purchased time on media to determine quantity, quality viewer ratings, etc. of the advertisers ads. Could use graphics to display this info.
10. For other clients we need more Media Analytics such as hit reports, viewership, audience numbers, length, tone, etc.
11. Improve security including making this website https:, 2-factor authentication & parameter checking in queries.
12. Add more error handling including "DB not currently available" problem. Note: All serious errors should be sent to the database for later analysis by the Administration app.
13. Improve the help functions. Menu explains to users how to use the search option, email alerts, edit, download, transfer of data etc. Update help page with ability to email questions, suggestions & surveys. Add proactive &/or contextual help to provide help without having to go to the help tab.
14. The search engine also scrapes the web for newspapers, magazines, and some web social media. Add additional web scraping and searches for podcasts, blogs and more web social media.
15. Media Analytics reports & graphs are partially completed but need a lot more work. (hits, viewership, audience numbers, length, tone (negative, neutral, positive), etc.
16. Language translation needs to search in multiple languages on request & translate results. This has worked in the code in the past using Google translate, but has become too expensive. For alternatives see: <http://3.13.169.109:4200/languagetranslation> Open Source Language Translation <https://www.goodfirms.co/computer-assisted-translation-software/blog/the-top-10-free-and-open-source-computer-assisted-translation-software#:~:text=Lokalize%20is%20a%20free%20and%20open%20source%20computer-assisted,the%20GNU%20General%20Public%20License%20%28GPL%29%20Version%202>.
17. Improve Results (hits) and Detail pages. Make reports more user friendly, possibly with graphic representations of some data. Make sure search words in results and details pages are always highlighted.
18. Finish work on the builtin video editor on the details page. The user highlights the text words for the portion of the video that defines what the user wants their new video segment to contain. This has been working in our preliminary demos. Be able to keep these new videos on the editing page where they can then be concatenated to produce a final video. The user should also be able to create video segments from the Previous & Next recorded segments. This is far enough along that it might be completed in a couple semesters. Note: this editing is accomplished using the ffmpeg API.
19. Update and improve report download file format (TV: mpeg, wmv, mp4, wmv, etc.) (Radio: mp2, wav, etc.) & resolution choices. This should work with Mac or PC. Implement better ways of downloading video & audio segments from the editor & in email alerts, something like [www.wetransfer.com](http://www.wetransfer.com). Could be called digitransfer.
20. Add additional media such as podcasts and cable TV.
21. Add ability to do press releases to journalists & other media.
22. Merge Github branches. Do a build version and put it up on AWS. **IMPORTANT:** Check costs of using any additional AWS Amazon Lightsail facilities that your code might need.
23. Add a Checkin through Administration that shows the app is running properly.

**ADMINISTRATION FRONTEND**: This site allows us to monitor and administrate the various machines on the digiclips network and display statistics about machine performance and analysis of data collected. There are some working graphics under the dashboard tab. Eventually, some of these statistics & graphics should be available to our customers on the Search Engine website. There are also some data analysis graphs that have been prototyped in Excel, but still need to be implemented in Administration. Administrators can use the Admin website to monitor the performance of the TV recording computers & the radio recording computers. This website currently runs on a computer on our network and in the cloud (AWS Lightsail). It can display errors encountered by the TV recording programs. It can also be used to administer users on the Search Engine website. It is written using Angular 12, Typescript, Javascript, Express, Node.js and React. We realize that it might be better to do statistical analysis, etc. in other languages, frameworks or packages and are open to your suggestions. Some standards we use are MP3, MP4, WAV, ASS, SRT, VTT. Some tools we use are Ffmpeg, GitHub, OpenVPN, SSH, AWS Lightsail and Chrome Remote Desktop.

Current menu structure:

Verify Accounts <http://3.141.93.97:4200/verifyaccount>

Dashboard <http://3.141.93.97:4200/dashboard>

Add on more media <http://3.141.93.97:4200/media>

Brand <http://3.141.93.97:4200/brand> - name change for system

MediaAnalytics <http://3.141.93.97:4200/mediaanalytics> Note: This should be expanded to Media & System Analytics.

Help <http://3.141.93.97:4200/help> - Explain system monitoring tools.

1. It is written in Angular 12 & needs to be updated to Angular 13. All node packages will also need updating.
2. It keeps track of all recording machines and devices on the system (<http://3.141.93.97:4200/dashboard>) but needs to include other machines & devices. Monitor Database machine (make sure MySQL daemon is running properly and efficiently).
3. Improve existing checkins (messages sent to the database indicating that the recording is running ok).
4. Fix Verify Search Media users & passwords (<http://3.141.93.97:4200/verifyaccount>) with email alerts sent to administrators for approval. Needs to be able to add different types of users like one time users or subscribers: daily, weekly, monthly, quarterly or yearly. Add ability to email credentials, etc. to new or old users. Later add in general public users supported by advertising dollars (low priority).
5. Expand by adding on more media <http://3.141.93.97:4200/media>. Use administration to add more stations, etc.
6. Improve Brand name changing and other abilities (<http://3.141.93.97:4200/brand>) NOTE: To change the name say from Digiclips to another name of who is using the rights to the software. Maybe we need a Setup facility for any new company that buys our software to change this and other parameters such as which media to track.
7. Improve Email alerts to administrators (that currently just shows if the machines are pingable and check in from their running programs) to administrate the problems on the machines.
8. Add a Super Administrator to approve lessor administrators and assign their permissions.
9. Add ability to maintain client and prospect lists.
10. Add the ability to change resolution and other recording parameters for Radio & TV in the config file. Improve & test additional configuration file changes in administration and record programs.
11. Security on the entire system including making websites https:, 2 factor authentication, etc.
12. Replace Ftp file transfer protocol with more secure protocol.
13. Research how to & Update our VPN link to use the new SSH3, we currently use OpenVPN on our router that uses SSH2.
14. Improve keeping track of all system machines computers, including existing checkins. Check Search Engine machines to make sure their SSH Virtual directories and VPN are all functional. Restore any that are not working.
15. Add ability to import external data from Nielsen and others.
16. Add more Administration analytics, additional charts and graphs, many which have already been prototyped in Excel, to keep track of entire system health and efficiency.
17. We would like to add more data analysis of the performance of TV and radio, audience numbers and media values over time, ect.
18. Improve notification emails to administrators to administrators such as DNRs (did not record), recording machine program crash/malfunction or machine not pingable
19. Add ffmpeg error tracking from TV & Radio recording machines and add error list & graphic analysis of errors.
20. Track signal strength of TV or Radio on air. Missed data could be with Hauppauge board, the broadcast station itself or with the antenna.
21. Adding or deleting media links like TV stations, Radio stations, newspapers magazines, podcasts etc. to existing recording machines. Add ability to add Program name from a TV guide like <https://www.titantv.com/Default.aspx?r=t> to capture video.
22. Improve keeping track of error messages with improved lists and graphs.
23. Add archiving & maintaining data with zip compression of data for long time storage.
24. Improve and update menu help. Update help page with ability to email questions, suggestions & surveys. Add proactive &/or contextual help to provide help without having to go to the help tab. Could even add context sensitive help in places.
25. Merge Github branches. Do a build version and put it up on AWS. **IMPORTANT:** Check costs of using any additional AWS Amazon Lightsail facilities that your code might need.

**TV BACKEND:**

Uses API software [www.ffmpeg.org](http://www.ffmpeg.org) also written in C. We currently use these stations for development: KCEC - KDEN - KMGH - KTVD - KWGN - KUSA - KDVR - KCNC. They are double captured for backup purposes. The TV Backend records TV stations 24/7. Each backend machine records several local over-the-air TV stations, keeping 10 minute segments of overlapping video overlapped by 2 minutes, audio, video and closed captions are archived locally. Each machine also sends a copy of the closed captions in smaller overlapping chunks to the database server for searching by the SE app.

1. Through Administration be able to add on more Television stations from free on air antenna and streaming channels using the config file, database & Admin. (partially completed)
2. Send signal strength of TV on air to Database for later analysis by Administration website.
3. Send more ffmpeg errors to the database by extracting information from ffmpeg error logs.
4. Closed caption doesn’t get all the data for searching, so we are in the process of adding video-to-text and audio-to-text but still need more teams to work on this. Possibly also add Image Recognition. Does need a lot more work. Note: Audio-to-text is in stereo left and right channels.
5. Phonetic checker
6. Punctuation checker
7. Spell checker
8. Grammar checker
9. Language translation
10. Needs Lip Read to text (low priority)
11. Build a TV scanner/tuner to determine what channels are available and what their parameters are. We currently use a combination of several open source apps written in C to do this manually.
12. Checkin to database is used by Administration to verify that process is running is already implemented but needs improvement. Add a DNR count to the check in.

**CABLE TV BACKEND AND OTHER STREAMING VIDEO**

Not yet implemented lots of work needed here. Could be implemented as an addition to the existing TV Backend, but might be better as a separate backend. Needs development through the web using <https://watchnewslive.tv/> . Used to develop CNBC <https://watchnewslive.tv/watch-cnbc-live-stream-free-24-7/>. Administration should have the ability to add more Cable TV stations. Needs to send ffmpeg errors to the database. Will eventually need video-to-text and audio-to-text. Note: Audio-to-text should be in the stereo left and right channel. Will eventually also need Phonetic checker, Punctuation checker, Spell checker, Language translation, Video text recognition, Image Recognition & Lip Read to text (low priority). Use API software [www.ffmpeg.org](http://www.ffmpeg.org). Will also need a Checkin process.

**RADIO, SIRIUS & PODCAST BACKEND**

Is written in C and records KOA AM - KHOW AM - KCFR FM PBS and produces speech-to-text which is copied to the database. The app records 10 minute segments with 2 minute overlaps.

1. Add podcasts (not yet implemented).
2. Add Sirius Radio (not yet implemented).
3. Work with Administration to be able to add more radio stations & podcasts.
4. Speech-to-text in stereo left and right channels newer versions being implemented, but needs more work and testing.
5. Add backup double capture in separation geographic locations.
6. Add send signal strength of on air Radio to Database for later analysis by Administration website app. (Low priority because we use mostly streaming audio.)
7. Add extracting ffmpeg errors from ffmpeg error logs and send them to the database for later analysis by the Administration website app.
8. Phonetic checker
9. Punctuation checker
10. Spell checker
11. Grammar checker
12. Language translation
13. Checkin to database is used by Administration to verify that process is running is already implemented but needs improvement.

**DATABASE & ARCHIVING**

1. Add new or improved video-to-text, audio-to-text tables, archive tables.
2. Currently TV, Cable TV and Streamed TV can be in Hi Res and archived for only around 3 months on recording machines, but text from these recordings is kept for much longer in the database. So we need to implement compressed data in zip and other ways of storing video such as archiving TV in SD or lower quality for longer term archives. Archived data should cover TV (Cable TV Streaming), Radio, online Newspapers, Magazines, Social Media, Blogs, etc as needed.
3. We also work with Track TV [TrackNEWS Library (itracknews.com)](https://itracknews.com/) that has archived media data back to around 2003. Eventually we will want our programs to be able to search and import this older data when relevant to a search. This is currently done manually.
4. Tone, sentiment, polarity for text (is currently being worked by Regis University).

**INFO WEBSITE**

The current website written in an old version of Angular gives information about DigiClips and used to allow customers to contact us by filling out a form. A newer version written in Angular 9 has not yet been completed. Among other improvements it needs a backend to allow emailing to function again. Add a customer survey to the website with results sent to the database. It should be updated to Angular 13.

**GENERAL INFO**

**CLASS TEAM AGENDA**

If you send Henry and myself your resumes we can better advise you on what projects you would probably like and be most successful at. **Feel free to email or call us for more detailed information on possible projects.**

Choose a team leader to do general communication with Henry & Bob (like meeting agendas etc.)

Divide up the project with the teams and the team members.

We do Online Google Meetings or Zoom once a week. Email us the meeting agenda day before.

Screen share in meeting what you are working on

We will email back notes after our meetings.

Email to Henry and Bob: Put DIGICLIPS <School Name> in the subject line so we can filter your email to proper location.

Henry’s email hbremers@gmail.com

Bob’s email bobshapiro40@gmail.com

Henry Cell (303) 249-4676 or Bob (303) 926-0334 or Cell (720) 280-5335

Please ask us lots of questions and keep us informed.

Get on DigiClips github and study the code & comments developed in the past.

Work on your plan to solve the problems your project has identified & submit to Henry and Bob for review.

Some problems may be solved very quickly, some will take much longer, maybe even years with multiple teams.

**IMPORTANT**: All frontends need to be developed in Angular 13 or updated see: <https://angular.io/> to learn Angular online or as a reference.

**REALLY IMPORTANT**: **Allow 30 to 45 days before the end of semester for testing, debugging & improvement feedback** from Bob/Henry before you finish your project and do your class presentation.

Make comments in the code as to what it does, where borrowed code came from and suggestions for needed improvements (TODOs).

**First** Allow for 30 to 40 days before the final presentation to demo to Bob & Henry to identify and fix any problems.

**Second** Do a readme file explaining the entire process needed to set up the app. Put it all up on **DigiClips Github (not your Github account)**.

**Third** For final frontend presentations put it up on AWS Amazon Lightsail.

**VERY IMPORTANT**: Be aware of Cost of using AWS Amazon Lightsail. AWS not to be used for processing or storing the data. Data should be collected and processed on the DigiClips network machines.

Search Media: Put on AWS Amazon Lightsail <http://3.13.169.109:4200/>

Administration: Put on AWS Amazon Lightsail <http://3.141.93.97:4200/>

**SOFTWARE TO BE FAMILIAR WITH**: Ubuntu-Mate 20.04 or 22.04, C, nodejs, react, angular 13, typescript, MySQL relational database, AI Artificial Intelligence APIs.

**AWS AMAZON LIGHTSAIL**

AWS Amazon Lightsail **IMPORTANT**: not to be used for processing the data or storing the data. It should be used for the front end presentations connecting directly to the processed data on our network. Data processing should occur on our machines. ProjectsPP

Search Media: Put on AWS Amazon Lightsail <http://3.13.169.109:4200/>

Administration: Put on AWS Amazon Lightsail <http://3.141.93.97:4200/>

Informational Webpage: To be put up on AWS Amazon Lightsail in the near future.

Bob Shapiro

bobshapiro40@gmail.com

(303) 926-0334

Henry Bremers

hbremers@gmail.com

(303) 249-4676