

DigiClips Media Monitoring

FINAL REPORT

Team: sdmay23-03

Client: DigiClips Inc.

Advisor: Dr. Ashfaq Khokhar

Team Members:

Aashish Komaragiri	- <i>Team Engagement Officer</i> - <i>Database Analysis</i>
Elijah Shelby	- <i>Testing Officer</i> - <i>Database Analysis</i>
Grace Rasmussen	- <i>Team Organization Officer</i> - <i>Search Engine Application</i>
Khushveen Kaur Umra	- <i>Advisor Interaction Officer</i> - <i>Search Engine Application</i>
Matthew Karmelich	- <i>Client Interaction Officer</i> - <i>Email Automation</i>

Email: sdmay23-03@iastate.edu

Website: sdmay23-03.sd.ece.iastate.edu

Table of Contents

1	Revised Project Plan	3
1.1	DESIGN EVOLUTION	3
1.2	FUNCTIONAL REQUIREMENTS	5
1.3	NON-FUNCTIONAL REQUIREMENTS	6
1.4	STANDARDS	7
1.5	ENGINEERING CONSTRAINTS	8
1.6	SECURITY CONCERNS	9
2	Implementation Details	10
3	Testing Process and Results	11
4	Work in the Context of	11
4.1	RELATED PRODUCTS	11
4.2	RELATED LITERATURE	11
	Appendix I - Operation Manual	12
A1.1	STARTING DIGICLIPS SERVER	12
A1.2	DEMO INFORMATION	12
A1.3	TESTING	14
	Appendix II	15
	Appendix III	16

1 Revised Project Plan

1.1 DESIGN EVOLUTION

For the most part, we were able to stick to our original project plan. We followed an agile project management style along with the use of task decomposition. This helped to keep us mostly on track throughout the semester.

There are two main subparts for our project, which are as follows:

I) Automated Email Reports:

The automated email reports might be built differently for each client using an agile strategy, as each client may have different preferences. As a result, we aim to address this topic with a broader viewpoint. To handle this challenge effectively, we'll need a variety of perspectives.

Thus, we are certain that we can overcome the hurdles presented by this assignment using an agile strategy. This also means that we will need to break down and reorganize the team assignments, as we progress through the semester.

● **Tasks in Automation Email Reporting:**

- Scraping Web Data on text client want searched
- Sentiment Analysis on Positivity of Media
- Database connection to store media, or get client text information
- Converting data found from the search engine, in a readable format to client
- Figure out a way to automate email sending to client, once data is converted

II) Graphical Representation of Data:

These graphical representations of data might be structured differently for the clients or companies we deal with using an agile strategy. Again, from an adaptation standpoint, we must be prepared to account for this. When additional data becomes available in the future, this is another area where adaptation will be required. Because agile accommodates for adaptability, we feel we may adjust the project scope when we encounter certain instances. As a result, the overall project aim is only hampered a little.

We are aware that there would be several obstacles in the way of completing these difficult duties. As a group, we believe Agile is the best, and the most efficient way to approach software challenges like these.

- **Tasks in Graphical Representations of Data:**

- Find APIs or libraries that can be used to create easy, readable graphs that the client can use for their own use
- Store audio, and video clips pertaining to clients request to DigiClips' database
- Develop graphs specific to the clients request, based on data obtained from media segments
- Adapt for moving requests, when companies want different data points
- Create a potential analytical report stating the trends that were found in the graphs

III) Frontend Email Alert Options Webpage:

We created a new and improved webpage where clients can go to manage their current email alerts, as well create new alerts that they may want. This includes a simple form that requires users to set their preferences on each individual report. They are able to choose their keywords, media types, and dates when it comes to the media we search, as well as the format, frequency, and number of results included in each alert.

1.2 FUNCTIONAL REQUIREMENTS

I) Automated Email Reports

This project will look into existing solutions and create efficient automated email modules that will present the findings from various media sources depending on the keywords and phrases of interest in these segment recordings. This project will focus on upgrading the email alerts setup page through extensive field testing, allowing customers to request certain sections of the recorded segments depending on their preferences.

The **functional requirements** for the automated email reports are as follows:

- Create an automated reporting tool that conducts automated searches on the DigiClip's database to alert their clients when important information regarding their queries is retrieved
- Collect all new media information regarding searched keyword and/or phrase and send all relevant information to the respective client
- Create a readable format for the email being sent, such that retrieved information is presented with correct links to respective media sources
- Allow for easy transitions when clients want different types of data displayed
- Improve the "help" functionality in the email reports by adding proactive and/or contextual help to provide help without having to go to the help tab
- Organize the retrieved information in a more user-friendly report

II) Graphical Representation of Data

This project will look at existing solutions and create effective graphical data representations for the DigiClips web dashboard, which will be available to DigiClips administrators and their clients. The analytical components of the results revealed from the search engine database when a customer searched for keywords or phrases from different sources of data will be included in these graphical representations of data.

The **functional requirements** for the graphical representations of data are as follows:

- Retrieve data from media segments being searched
- What type of data are we looking for?
 - Viewer ratings on certain keywords wanting to be searched by the client
 - Sentiment analysis on the positivity or negativity towards certain keywords
 - Length of the requested media segments (Videos & Audios)
 - Competitor / Current client ads

We did not require any other physical resources to complete this project. In terms of the digital resources, we had access to the existing code that DigiClips has pertaining to automated email reports and front-end search engines. We also accessed the DigiClips database, backend code, and company VPN, in order to ensure that we are able to integrate our solutions with the rest of the DigiClips Media Search Engine.

1.3 NON-FUNCTIONAL REQUIREMENTS

General Resource Requirements:

The resource requirements will be used for both the parts of the projects, as these requirements will give us access to the previous technical documentation. The general resource requirements are as follows:

- **Github access to current production code**
 - We will need access to DigiClip's production code to integrate the changes from this project
 - We will need access to GitHub to update the technical documentation for this project
 - We will need access to GitHub to update the Angular version from Angular 12 to Angular 13 for their production code

- **Access to Company VPN**
 - We will need VPN access to allow us to access the company portal to access important information and tools to assist in completing the project
 - We will need this access to see the workings of DigiClips to improvise the email reports, and the options page accordingly

Physical Requirements:

The physical requirements will be used for both the parts of the projects, as these requirements will allow us to implement the technical requirements for this project. The physical requirements are as follows:

- Regular communication between our group, the client, and the advisor
- Maintain proper technical documentation throughout the semester, to avoid any miscommunications
- Update the Angular version to keep the production code up to date

Aesthetic Requirements:

The aesthetic requirements will be used for both the parts of the projects, as these requirements will allow us to work on the user interaction for this project. The aesthetic requirements are as follows:

- Create user-friendly automated email reports, to easily organize the retrieved information from the database based on the search criteria
- Follow the general layout of other pages completed during the previous projects
- Displaying email alerts in a clean, readable format for companies to use
 - This can help with clients to utilize information set to them in a much more easier way
- Displaying graphical information in an organized, easy to read format
 - Allows for clients to understand comparisons being made between data points
 - Allows us as administrators to keep a record of these data points in a much more efficient way, so as an administrator there can be additional features that can be implemented in the future regarding this data

1.4 STANDARDS

In this project, all standards and practices will be software focused. Specifically, for this project we will be using practices such as object-oriented programming as well as the following IEEE standards for software development best practices:

- **IEEE 830-1998:** Recommended Practice for Software Requirements Specifications
 - This practice is broken down into five clauses
 - Clause 1 explains the scope of this recommended practice.
 - Clause 2 lists the references made to other standards.
 - Clause 3 provides definitions of specific terms used.
 - Clause 4 provides background information for writing a good SRS.
 - Clause 5 discusses each of the essential parts of an SRS.
- **IEEE 829-2008:** Standard for Software and System Test Documentation
 - The testing process is used to determine whether certain development products of a task follow requirements.
 - Also test whether the software satisfies given requirements of the user
 - The testing scope is really big
 - Software-based systems
 - Computer Software
 - Hardware

- Interfaces
 - This process is used for software that is being developed, maintained or reused.
- **IEEE/ISO/IEC 26531-2015:** International Standard for Systems and software engineering -- Content management for product life-cycle, user, and service management documentation
 - This standard gives requirements for efficient development and management of content produced throughout a project.
 - Our contributions to this project will need to be organized and easily accessed by other teams, our clients, and each other.
 - Everyone needs to be able to easily understand any additions or changes that have been made.
- **IEEE 1063-1987:** Standard for Software User Documentation
 - This standard provides minimum requirements on the structure and information content of user documentation
 - It addresses editorial and stylistic considerations only when they impact structure and content
 - Editorial and stylistic considerations are addressed only when they impact structure and content

1.5 ENGINEERING CONSTRAINTS

- Must follow various copyright laws that might be relevant when displaying clips and snippets of different media types
- System should be built without utilizing any costly APIs / cloud resources
- System should be built with documentation to explain usage and integration
- The developed system should be computationally efficient and able to run on a relatively underpowered computer
- The program should be able to operate quickly enough for customers to query data within 24 hours of recording

1.6 SECURITY CONCERNS

Since so many students have access to the DigiClips codebase and their servers, there is a cybersecurity concern. As a countermeasure, there should be strong password policies and two-factor authentication to prevent unauthorized access to the system. Encryption should be used to protect sensitive data both in transit and at rest.

If the email reports and media segments will be accessed from mobile devices such as smartphones or tablets, they need to ensure that the devices are secured with passwords, biometrics, or other security measures. All users should be trained on how to keep the devices physically secure and how to report lost or stolen devices.

Since the graphical representations are also linked in the email reports, if the graphical representations contain sensitive data, we must ensure that they are stored securely and that access is restricted to authorized users only. We have considered using digital rights management (DRM) technologies to prevent unauthorized copying, distribution, or modification of the graphical representations.

2 Implementation Details

Required Technology

- NodeJS and Node Package Manager (NPM)- NodeJS is a runtime environment that executes JavaScript code outside a web browser, while NPM is a package manager that allows easy installation and management of packages/libraries for NodeJS applications. The email alert management is done using this language.
- Angular- A popular open-source JavaScript framework used for building single-page web applications. It provides a powerful set of features for building complex and dynamic web applications which our frontend utilizes. Angular is faster yet uses less code.
- OpenVPN Connect- OpenVPN is a vpn network that connects two endpoints over the internet. OpenVPN allowed us to connect one and view the database and connect to a remote desktop from our local computers.
- MySQL Databases- This holds all the information that is important to the clients. Whenever a hit is found, it is placed in the database. Newspapers and magazines update automatically when the project is deployed and stored in the database for at most two weeks.
- Amazon Web Services (AWS)- is a cloud computing platform that offers several cloud services. This project utilizes AWS lightsail for both the frontend and backend. AWS Simple Email service (SES) is the environment used to send emails automatically.
- Remote Desktop- is a remote access technology that allows users to connect to a computer from a remote location. This was used to connect and work with the frontend machine.

Coding

- Github and Git Bash- were used to manage and share the code, work on specific aspects to our project without altering the main system, and collaborate with others. Git bash was the command line interface that was used to communicate with git.
- Visual Studio Code- is the code editor platform that was used to edit our code
- AWS SDK- is used for JavaScript. It is a collection of libraries and tools that help with building browser-based/server-side applications that interact with AWS services
- Postman- was used to generate html reports that were sent to the AWS services which is then sent to a specific email.

3 Testing Process and Results

In our software development process, we employ a comprehensive testing approach, utilizing Node.js, Cypress, and Jest as the primary tools to ensure the highest quality of the DigiClips codebase. Node.js provides a robust and scalable runtime environment, while Cypress and Jest are employed to create and execute end-to-end and unit tests respectively. Through this combination, we have achieved 100% test coverage, which means that every line of code, function, and component within what was worked on has been thoroughly examined and verified to behave as intended. This level of test coverage allows us to maintain a high level of confidence in our software's stability, reliability, and overall performance, while minimizing the potential for unforeseen issues or regressions.

Further manual end to end tests were required to ensure emails were formatted correctly. This required repeatedly sending emails with different inputs and formats to ensure that the output always looked as intended. This was not ideal, as the tests are not easily repeatable, but we were unable to find a better solution.

4 Work in the Context of

4.1 RELATED PRODUCTS

Currently, most digital media monitoring services use outside software in addition to their own tweaks and additions in order to provide their clients with information. With DigiClips, the goal is to build everything themselves from the ground up. Although this will be much more difficult and time consuming, in the end it will allow for the capability of making things much more customizable depending on various clients' needs.

4.2 RELATED LITERATURE

There is not much related literature on this topic as everything is fairly consistent and smooth sailing in this type of work. Not many people need media monitoring. The majority of clients are going to be advertising companies or people who get unnecessary bad publicity that could negatively affect their careers in the long run (lawyers, actors, professional sports players, etc). Those that do need media monitoring or like to use it for business trends have found a solution in other companies that works fairly well.

Appendix I - Operation Manual

A1.1 STARTING DIGICLIPS SERVER

Installations required to start DigiClips Application locally

1. OpenVPN - DigiClips is based in Colorado, so to be able to access their system we will need a VPN connection
 - a. Once OpenVPN has been installed you will be given a .ovpn file which will give the important information that is needed to connect to the VPN
2. mySQLWorkbench - When starting the server, and working on the code we will need to connect to the DigiClips database server
 - a. To access the code through the server you must be an authenticated user
3. DigiClips App Repository - You will need to request access to obtain permissions to read/write to this repository
4. 2 Terminal Windows
 - a. One to compile the DigiClips application
 - i. Cd into the main directory of the repository
 - ii. Npm install to download all the latest packages
 - iii. Npm start
 - b. One to start the DigiClips Server
 - i. Cd into the server directory of the DigiClips repository
 - ii. Npm install to pickup all the packages that are needed to run the server
 - iii. Npm run debug to start the server, and run on debug mode so we know when there are problems occurring on the server

A1.2 DEMO INFORMATION

1. Automated Email Process for Newspaper Alerts

The first part of the demo will showcase how we were able to take keywords, and other search parameters to conduct a search over newspaper feeds and collect relevant client information.

- AWS Email Sending setup
 - We as a group has to create a AWS account
 - Setup configuration like region, permissions in email sending
 - Verify emails that are valid for sending/receiving emails
 - Implement the code to send a DigiClips Newspaper Alert

- HTML Service Setup / Connection
 - Created a service which handled requests and develops a HTML automated string that can be sent to AWS
 - This HTML String will be utilized and displayed on email
- Stored Procedures to pull newspaper data
 - We had to be authenticated beforehand to access/utilize the stored procedures.
 - Stored procedures were used to pull newspaper results from the database, and organized in a easy readable format for the client
- Created a Angular Component which is responsible for creating handling Email Reports

2. Email Alert Form

- Created a Angular Component which is responsible for creating handling Email Alerts
 - This component takes information that is shown from the HTML, and sends to the SQL Database

3. Angular Upgrade

- Updated entire search engine front-end from Angular 13 to Angular 14.
 - This allowed for a more secure, concise, and efficient web application.

4. New HTML Format for handling search results for DigiClips

- Created an angular component that is used for the newly designed email reports
 - This Angular component handles all information related to the newly designed email report system
 - The HTML report is able to handle request being sent
 - Once a search request is initiated, the newspaper search occurs
 - Results are displayed in a way that is easy to navigate for the client

A1.3 TESTING

The application consists of two test suites

Server Tests

Required technologies:

- Node.js
- Jest

How to run tests:

From the root directory, enter SE/server/
Enter the command npm test

Front End Tests

Required technologies:

- Node.js
- Jest
- Cypress

How to run tests:

From the root directory, enter SE/src/
Enter the command npm test

Appendix II

Email Report Design

- Originally what we thought for the email report design is to have graphs to resemble certain statistics for the client's media search.
- The client had a change in direction for that they wanted a report to look like
- They wanted a list of the hits -> newspaper, social media, articles etc... displayed with links of the hits
- This would allow for a much more organized approach when the client wants to view an email report.

Sentiment Analysis

- We wanted to create a sentiment analysis for hits that were generated, and shown
- This deemed to be a very hard, and unpractical task for us to complete before the end of the semester
- The DigiClips project was developed using Angular - none of us in our group have been used to using Angular so this challenge in itself really slowed the progress we wanted to make for this project

Addition of Cron Job for Email Alerts

- This is a task that we really wanted to complete, mainly because we wanted to connect the Email Alert form component to an automated schedule.
- This would allow for emails to be sent in frequent intervals, and really complete the task that we have at hand

Appendix III

Knowledge of Angular

- Something that was very unfortunate during our senior design this semester is that we had to use Angular to develop the frontend application
- None of our group had any experience when it came to Angular, so understanding how components work, the syntax and way it connects to other files took a bit of time
- After this semester, we as a group definitely think that we grew as front end developers and think we can handle challenges when it comes to Angular

Working with another DigiClips team

- For half of the semester we were able to connect with the Colorado team that works under DigiClips
- They had some insight on some things we were implementing with the Email Reports
- We were both able to collaborate and develop connections like you would do in a real Software Engineering job