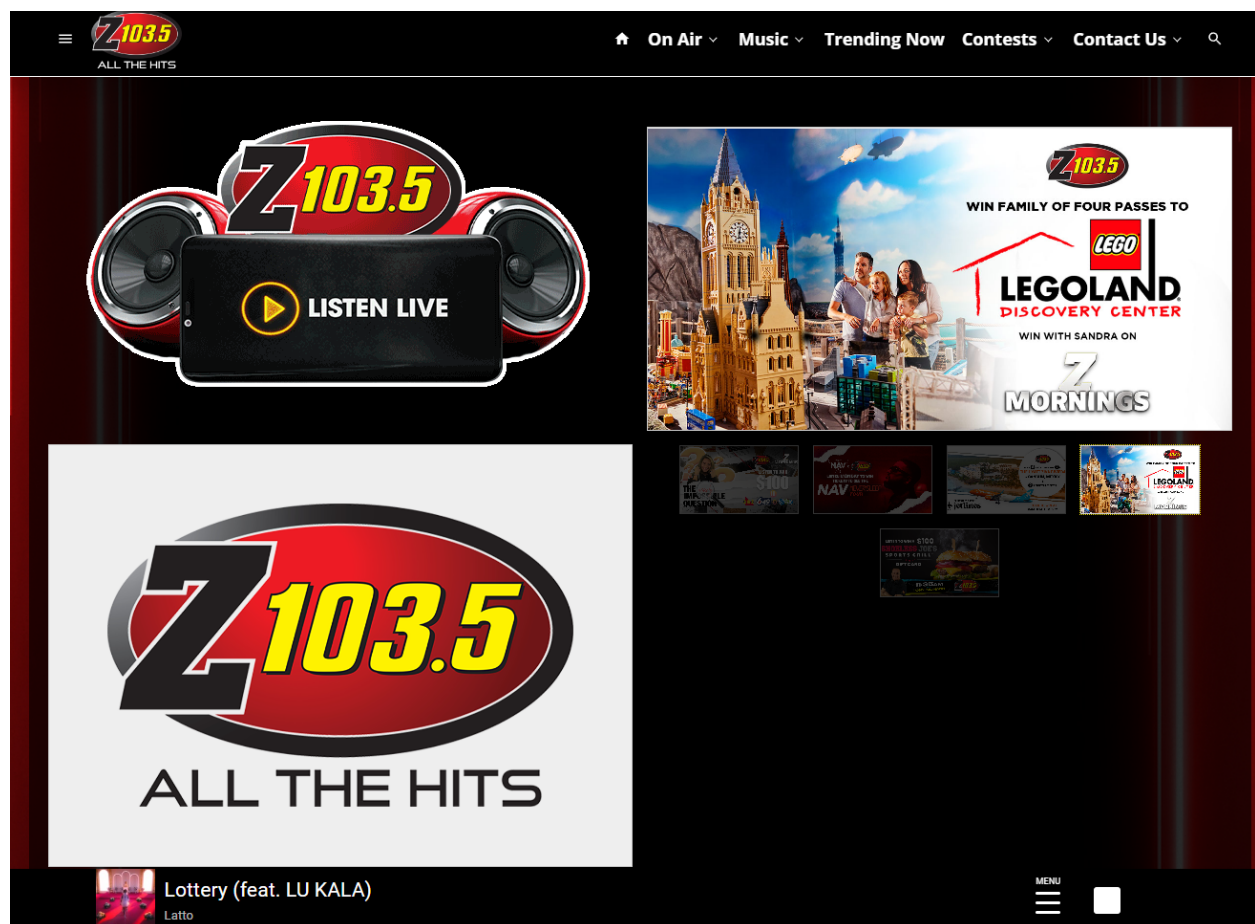


Student Name: Shavin Singh  
Student Number: N01569183

## Capstone Project Pitch: BrickMMO City Radio

### Brief Description of Project:

The goal of this project is to create a 24/7 Ai-driven Web-based radio broadcast based on the BrickMMO Lego city. The radio station will stream its content to a web interface, where anyone can visit it from their browser, and listen to an AI voice read news stories, weather, and other happenings around a fictional city.



Current Radio website with “listen live” and live player

The radio broadcast will have many of the features of a current real-world radio station, in that it will support segments. This could include music, games, contests, and even audience interaction. Listeners would be able to interact with these segments through the website (take part in polls, play games, etc).

As a developer, this application will focus on showcasing the ability of generative AI models and how they can be used in tandem to create engaging content. It will also highlight my familiarity with technology such as CMS development, database security, and API integration.

## How it works

A user can visit the website of the broadcast, and listen to an audio live stream on the website. Users can also interact with the website, by creating an account and logging into the web portal. Users will be assigned the role of “Reporter”, where they can submit different types of content (stories, games, segments, etc) in the web portal. They will be able to manage their content in the portal. When they submit an item, it is added to a queue. A system admin will need to approve the segment before it is published.

When it is time to publish, the content will be sent to ChatGPT to generate a script. The script is saved and will be read out on the broadcast by a text-to-voice API.

A sample script has been provided at the end of this document.

## Planned Technologies

To build this system, I plan to use PHP Laravel, for both the backend CMS (reports and segments), as well as for the front end (radio live stream). I plan to use PHP as this system will require a lot of backend operations and Server-side rendering, which PHP in my experience works the best for. It will also reduce the number of resources needed to host the application. This will improve the performance of the site and reduce points of failure (such as a Node server failing/eating too many resources)

For APIs, I intend to use OpenAI's ChatGPT API for creating reports/news stories as it is the leading application for generative models currently. For text-to-speech, I will need to implement Voice RSS API or other TTS language interpreters. I will also integrate the BrickMMO API to interact with the Lego BrickMMO city.

Hosting will be done on a web server (shared or dedicated), at a domain that will be representative of the app (ex. <https://radio.brickmmo.com>)

## Describe how you think you will build it:

To build this project, I will have a single Laravel instance running on a dedicated server to provide it with the resources necessary for an extended uptime. I will develop a dashboard and login area for reporters to submit their content/interact with the broadcast, through reports or segments. There will be two roles that interact with the system, Reporters and Producers. A Reporter will have the “public” role, where anyone can apply to be a reporter and submit content. A Producer will be assigned by the site admin and will have elevated privileges over a Reporter. In addition to the functions of a Reporter, a Producer will be able to manage content and approve content posted by a Reporter. This signifies that the content is appropriate for the broadcast and is ready to be pushed live.

Once a report is approved, the request will be sent to chatGPT. Before it is sent, it will be packaged in a way so that chatGPT knows to respond with a script that a radio broadcaster would read. This can be done using chatGPT's "system" role. The report will be sent with the rules, and once the response is returned, it is stored in the "segments" table of the database. Once per day, the application will query the segments table, and build a broadcast that can be used for that day based on predefined requirements (ex. 10 music segments, 15 news, 5 game segments). Segments will repeat if there is not enough to fill a day. A Producer will be able to mark a report as "Breaking" meaning it will be added to the day's rotation immediately. After the selection of the segment, the script will be forwarded to VoiceRSS API, to retrieve an audio stream that reads the script out, and will store the recording in a table called "broadcasts". The audio files will be streamed through Laravel's audio player on the public-facing web page.

## User Narratives

### Listening to the Radio

A user can visit the broadcast website (ex. BrickRadio.com), and click "play" on a web player. The Audio stream will play from its current live position.

### Adding a Report

A user can select "login" from the broadcast website, and register for an account. Once registered, they can go to "Reports" and an interface will appear, allowing them to write content using Rich Text, and submit a cover image. Once they are done, they can click "Submit for Approval". They will receive a confirmation message, then be redirected to their dashboard

### Modifying a Report

In the Reporter dashboard, users can click "View My Reports". They will be able to see a list of reports they have submitted but have not been approved as yet. They can click "Edit Report", and make changes to that report.

### Deleting a Report

In the "View My Reports" screen, they will have the option to delete a report that has not been approved yet. They will receive confirmation before removing it.

### Creating a Segment

In the Login Dashboard, a user can go to "Segments" and add a new Segment. They will be given a predefined list of segment types they can choose from (trivia, game, song, advertisement). The user selects the segment type, and the form will populate with fields that

are appropriate for that type. Once submitted, they can review all of their segments that have not been approved yet, and Modify or Delete them as necessary.

Provide a user narrative that describes the most common task an admin user for this project would complete:

### Approving a Report/Segment

A Producer can log in, And view “Review Reports”. A list of unapproved reports will appear for them to read. After reviewing a report and making any changes necessary, they can choose to “Approve” or “Reject” a report. Approved reports will be sent to a “Queue” tab, where the Producer can review the current Queue and make last-minute changes.

## Appendix A: Sample script by ChatGPT

**Prompt:** "The building at 12 Grimmauld Place opened recently to let a group of wizards in"

Emmet: Good morning, Lego City! This is Emmet bringing you the latest news on our Lego radio station. Today, we have breaking news from the wizarding world. The building at 12 Grimmauld Place has recently opened its doors to a group of wizards.

Brick: Whoa, wizards in our city? That's amazing! Do they ride brooms and cast spells?

Emmet: Well, Brick, I'm not sure about that, but what we do know is that 12 Grimmauld Place has a rich history in the wizarding world. It was the ancestral home of the Black family, and later became the headquarters of the Order of the Phoenix during the Second Wizarding War.

Brick: Wow, that's fascinating! I bet they have some interesting stories to tell.

Emmet: Absolutely! According to our sources, the group of wizards who have moved in are members of the Wizengamot, which is the highest court of law in the wizarding world. They are here to attend an important conference on wizarding law.

Brick: Wizarding law? Now that sounds like a complex topic.

Emmet: Yes, it is. And that's not all, Brick. The conference at 12 Grimmauld Place will also be attended by representatives from the Ministry of Magic, the International Confederation of Wizards, and the Magical Law Enforcement Squad.

Brick: Whoa, that's a lot of important people in one place! I hope they don't accidentally turn any of our Lego City buildings into frogs.

Emmet: (chuckles) I don't think we have to worry about that, Brick. The wizards are here for a peaceful conference, and we welcome them to our city.

[Closing music plays]

Emmet: That's all for now, Lego City. Stay tuned for more news and updates on our Lego radio station.

Brick: And remember, folks, always keep a wand handy, just in case.