

Tensor Hero Frontend

Team Members

Forrest Brandt
Junyi Cheng
Sam Stromberg
Claire Zhu

Faculty Advisors

Coye Cheshire
Niloufar Salehi

Table of Contents

Acknowledgements	
3	
Abstract	3
Clone Hero Background	3
Problem Justification	4
Preliminary Research	6
Feature Roadmap	7
Design and Prototyping	8
Ideation	9
Design Language	11
Final Design and Key Features	11
Landing Page	11
Level Generation	12
Chart Visualization	13
Charts Hub	14
Chart Details	15
How It All Works	17
Technical Implementation	
17	
System Architecture	17
Real-Time Progress of the Model	18
Chart Visualization	18
Downloadable Charts	19
Open Source and Maintainability	19
Findings from User Testing	
20	
Future Design Opportunities	21
Works Cited	22
Appendix	23

Acknowledgements

We would like to thank our two amazing advisors, Coye Cheshire and Niloufar Salehi, for providing guidance and encouragement throughout our capstone. We would also like to thank the individuals from the Clone Hero community who spoke with us about the experiences that brought them to this game. Their candidness and passion for the community was apparent from the beginning and inspired us in our project.

1. Abstract

Tensor Hero is a frontend application that hosts a machine learning (ML) model capable of producing playable Clone Hero charts from any song. Our team conducted user interviews and usability testing to identify pain points within the Clone Hero experience to ground our frontend development. Within Tensor Hero, users can upload and share songs, search for other community members' uploaded music, and learn about the ML model used to create their chart file. In future iterations, our team plans to add a capability that allows users to choose between a range of difficulty levels that is more suitable for their skill level, as most of the newly created charts produced by the charters within the Clone Hero community are only available for the hardest level. Our team works to demystify the ML model that produces these charts and provides transparency to an area too often labeled as "black box" algorithms. This community is passionate, and charters pour many hours into manually creating charts. The aim is not to displace this community but support them by producing an application that works to bring in new members and foster their interest in Clone Hero.

2. Clone Hero Background

Clone Hero is a community-run music rhythm video game where users play a guitar-like controller to mimic playing a song. The goal is for the player to use the controller to match the series of notes, or chart that resembles the music from the song, as closely as possible. As the name suggests, Clone Hero is, more or less, the exact clone of the

now defunct video game Guitar Hero. New content, in the form of charts, is created by community members called charters. Charters regularly collaborate with one another to create and release setlists of new charts. Charting one song can take hours, and each is created manually, often in a charter's spare time.

Both Clone Hero players and charters have leveraged video platforms such as YouTube and Twitch to create buzz around their content. These platforms help attract new members who either have played Guitar Hero many years ago or stumble across the game for the first time. Some charters live stream themselves, usually on Twitch, creating new songs which provide transparency into how charts are formed. In addition, top players on Clone Hero post videos completing difficult songs and have garnered quite an online following. One of Clone Hero's top players, Acai, has over one million subscribers on YouTube and regularly interacts with other community members through live streams on Twitch. While Clone Hero only began in 2017, it has developed a passionate community that continues to grow as it leverages social media platforms.

3. Problem Justification

Members of the Clone Hero community tend to be experienced, highly skilled players who grew up with Guitar Hero or have been playing the game for some time. They enjoy playing *Expert* level songs, the highest difficulty level, and setlists are often created with these players in mind. Most songs are released as part of a setlist. Often many charters collaborate to curate a setlist and songs are selected due to the charter's preference. Some setlists have all levels of difficulty for each song, but often setlists only comprise *Expert* level songs. Another option is to release songs individually, but normally they do not get the same traction in the community as setlists. Therefore, setlists provide the best method for reaching a wider audience.

Players heavily rely on charters to provide content and song selection. Paying a charter to commission a chart for a particular song has grown in popularity, but it veers from the type of community Clone Hero has built. In addition, finding content for specific songs

can be difficult. Setlists can be released on many different websites and can feel overwhelming for someone entering this community even before they begin playing a song - assuming a chart for that song has been created in the first place. The current system overly relies on charters to provide content and that content is catered towards expert players. While charters may sometimes accommodate multiple difficulty levels for the same song, interviewees noted that the process was non-trivial and time-consuming, and, since they had already charted the full song at the initial difficulty, comparatively boring.

In contrast, the console-based precursors to Clone Hero supported multiple difficulty levels for each song, and the progression of levels that became available to the user as they played was designed to introduce and improve player skill. As a decentralized community formed around an extensible application, rather than a self-contained commercial product, providing these onramps for new players is a perennial challenge.

Tensor Hero aims to provide a solution for users to not only generate a chart file from any song using an ML model, but also as a place to share and explore content created by other users in one location. Our frontend application hosts a ML model, created by another MIMS capstone team of the same name, that can take in any audio file and return a playable Clone Hero chart file. However, we want to be careful not to whimsically apply an ML solution without taking the community values into consideration. Charters spend hours creating each chart, and it is often done for free in their spare time. We aim to provide transparency with how each chart is created and use our application as a complement to the community and not something to displace it. We want to provide an avenue for current and especially new players alike. To give them the autonomy to create content without relying on charters for all the material.

4. Preliminary Research

Desk Research

While conducting preliminary research on the Clone Hero community and understanding the community dynamics and user profiles, we looked at the other rhythm games that have similar game functionalities as references. The goal of the desk research and analysis was to help us better grasp what are the general use cases and common features of rhythm games community websites, and additionally, how can we design Tensor Hero differently given it puts more emphasis on algorithm-generated charts, user acceptance, and chart visualization.

We looked into mainstream rhythm games OSU! and Beat Saber. After the analysis, we defined a few differences between Clone Hero and the other games and a few features and areas that can inspire us when designing Tensor Hero:

Differences:

- Clone Hero is locally-run and not connected to the online space, so we are not able to pull many stats from the player-end directly
- In Clone Hero, metrics and statistics for the players to compete based on are limited compared to the other rhythm games. For instance, Beat Saber has a global leaderboard.
- By looking at the most popular genres of music and observing their Discord communities, we could tell that Clone Hero's player base is English-speaking and local compared to the other games which are more global. Users of Tensor Hero might have a completely different mental scheme when it comes to online interactions and demands for this type of website.

Inspirations:

- A centralized chart or beatmap hub not only allows players to find playable levels for the music they like, it's also a great way to give credits to active community contributors and highlight their efforts.

- On top of having Discord communities, users still actively enjoy commenting on and interacting with each other on the community websites for more specific discussion regarding a chart or playlist.
- Beat Saber also has a machine-learning-driven level generator made by community members, which is validating for us to work on Tensor Hero.
- Providing details on FAQ and How-to pages can significantly reduce the learning curve for new users and answer their questions.

These platforms and designs gave us a great starting point for creating our own design, and allowed us to really look at the limitations and constraints we're working with when it comes to Clone Hero. We were able to make a realistic estimate of scope on what we could accomplish for the game, but still have the agency to be creative.

Generative User Interview

- How we see ourselves solving the problem – fulfilling community needs / solving problems

5. Feature Roadmap

While we view the primary objective of the site as allowing a user to generate a new chart or find a previously-created one to enrich their experience playing Clone Hero, we also identified other features that could expand the site's capabilities and fulfill additional roles for the Clone Hero player community.

1. Just as human charters often specialize in music genres, we would expect that different users of the Tensor Hero site will have distinct and distinguishable libraries of songs they have uploaded. As a result, end users may want to search the library of charts by uploader, not just by metadata from the music itself. This naturally extends to enabling users to create profiles, connect with or follow others, and search or notification of uploads by another user.
2. While our prototype has room for "related charts" when viewing an individual song, we have not explicitly defined what relations we are contemplating. Initially, this could be songs by the same artist; with similar genre, year, or other metadata

tags; or other uploads performed by the same person. However, in the longer term, this feature could be built out as a recommendation engine, considering the user's personal library of liked and downloaded charts, similarity on the basis of features in the charts (e.g., a particular technique), or even qualities of the music itself not captured in metadata fields but when the ML model parses the audio file.

3. While the charters we interviewed expressed satisfaction with the software they currently use to create and edit charts, building some editing functionality into our site could simplify the workflow for users who do not ordinarily make their own charts, or for experienced charters who only want to make minor changes to the ML-generated chart before publishing it. This could be particularly valuable for the multiple-difficulty-level use case, in which an Expert-level chart has already been produced, and Tensor Hero is being employed to adapt it for the other difficulty levels.

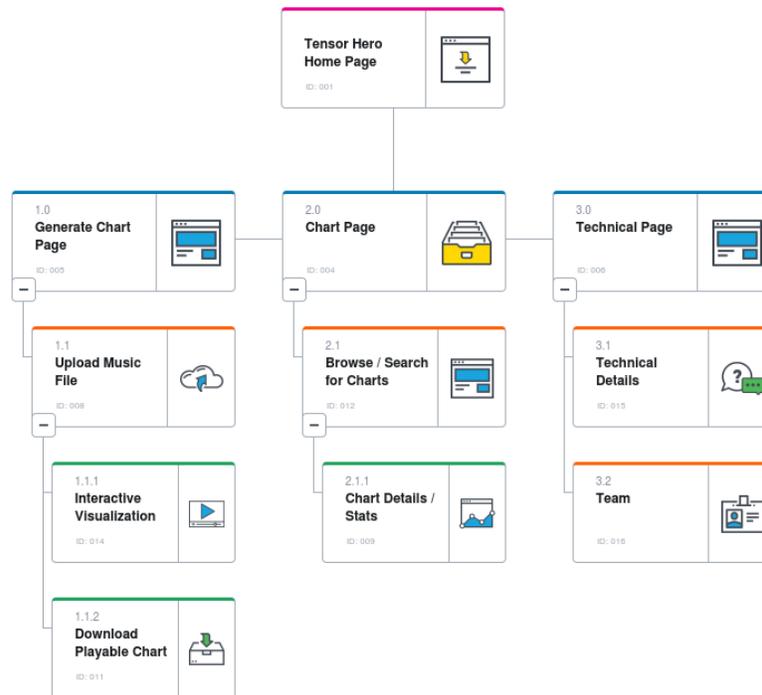
6. Design and Prototyping

Through our desk research and generative user interviews, and given the time limitations of the final project, we decided to design a product that bridges the gap between experienced players and inexperienced players when it comes to charting and chart playability; improve searchability of charts and allow the players to find music at one place; and finally provide a platform to cultivate more meaningful relationships among Clone Hero players. Through a few initial brainstorming sessions, we decided that Tensor Hero will be focusing on the following use cases:

- To help charters shorten charting time spent by providing an accurate foundation that's algorithm-generated
- For the less-experienced players, it's a great point of entry to generate charts for the music they like, or the easier levels for songs that only have expert or hard levels available
- To serve as an all-in-one hub for all Clone Hero charts

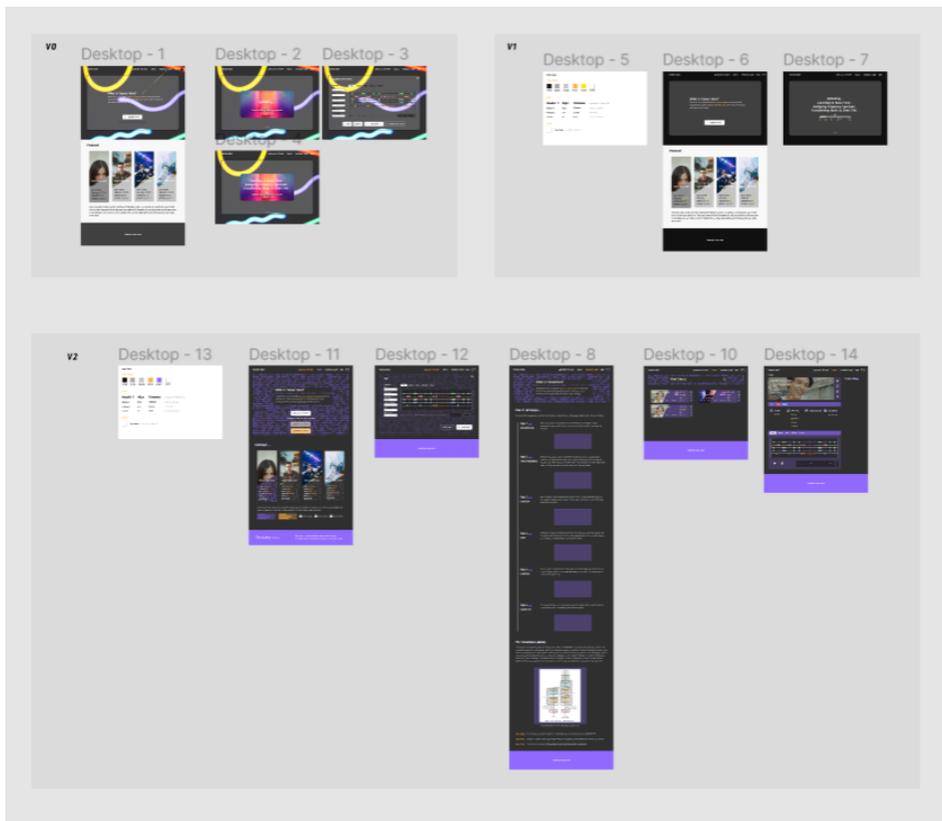
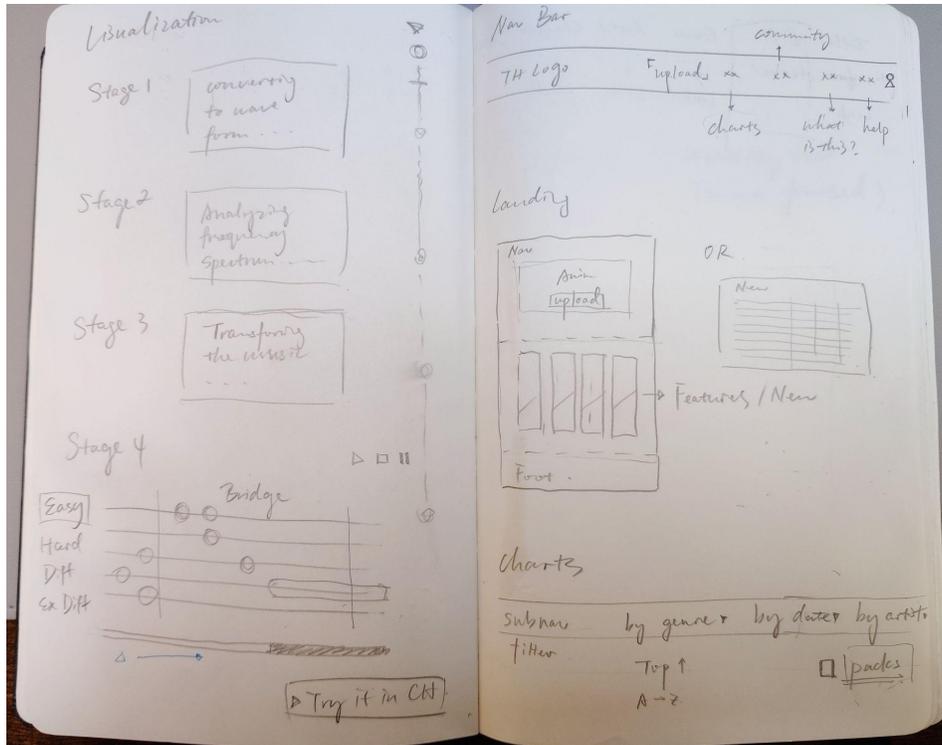
- To help users understand the technical background of Tensor Hero and incorporate Tensor Hero into their regular charting or playing flows

We proceeded to narrow those use cases down to specific page and feature areas, which also serves as the user flow of our website:



6.1 Ideation

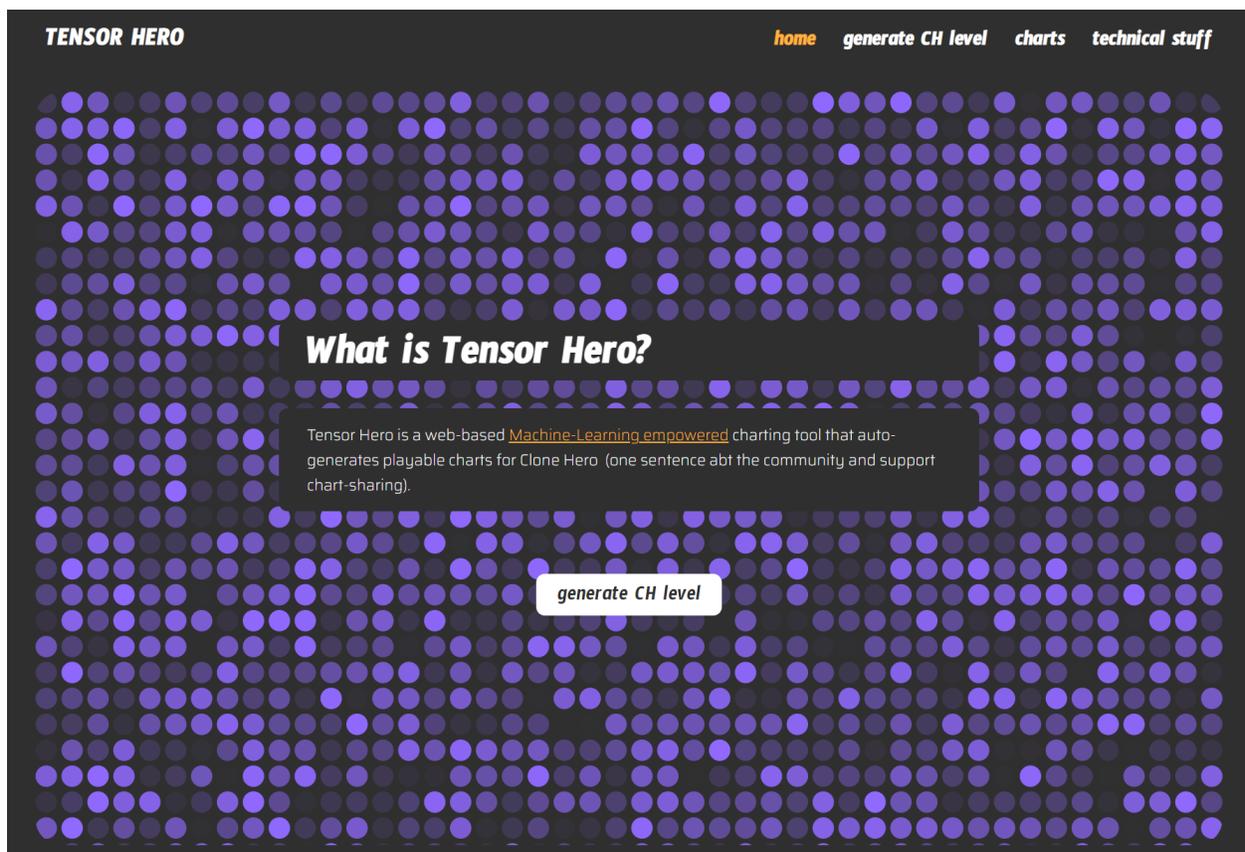
We took multiple iterations and mockups that led to the final prototype we have currently. After defining features, we first drew down sketches on paper to further guide our discussion on engineering feasibility and usability. We then translated those sketches into a low fidelity mockup, and iterated another version for it after discussing with our advisors. Finally, we finalized the design language and produced a high fidelity prototype that's the base model for implementation.



6.2 Design Language

Because the theme of Tensor Hero is around the concept of algorithm and machine learning, we want to keep the “feel” of the website simple yet techie. We chose the color purple to give that mysterious feeling, and also to compliment the dark overall theme. With orange being the color used for selection and text highlighting, it provides a pop of color and contrast that immediately catches the eyes of a user. White is used for regular text and call-to-action fields or buttons, and guides the users to where they’re supposed to click on the pages.

6.3 Final Design and Key Features



6.4 Landing Page

The landing page is kept high-level and minimalistic to allow the users understand what this website is about, and it is machine-learning empowered that’s different from the other charting tools. The dotted background “blinks” as the user stays on the pages to

make it more interesting to look at, which is also a visual element that's used throughout the website.

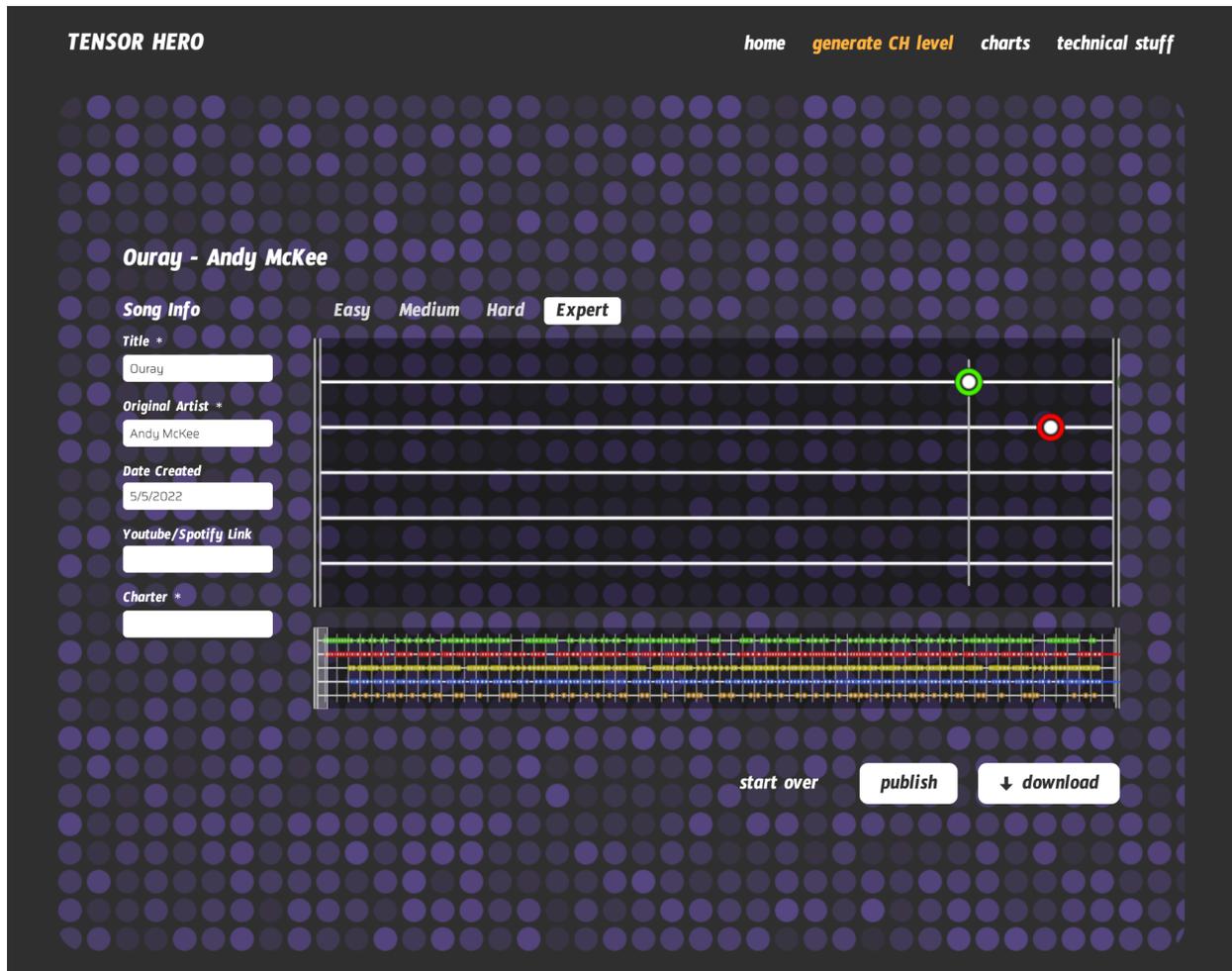
The navigation bar clearly states the other pages on the website. On the lower-mid of the page, there's a call-to-action button that directs the user to the level generation flow, which is the core user flow of the website, and the only major feature of the landing page.

The screenshot shows the 'Upload Music' form on the TENSOR HERO website. The form is centered on a dark background with a pattern of purple dots. At the top left, the text 'TENSOR HERO' is displayed. At the top right, a navigation bar contains the links 'home', 'generate CH level', 'charts', and 'technical stuff'. The form itself is titled 'Upload Music' and includes several input fields: 'Song Title *', 'Artist *', 'Genre', 'Album', and 'Year'. Below these fields are two file selection buttons: 'choose audio file' (with a note 'supported format: .mp3 .wav .ogg') and 'choose cover image' (with a note 'image preview will show up there →'). To the right of the 'choose cover image' button is a 'preview' button. At the bottom of the form is a 'Submit' button.

6.5 Level Generation

After clicking on the button, the user is taken to the “generate CH level” page and can start to upload a music file for chart generation. Since we need the song title and artist name for the model, we ask the user to input information about the song, including a cover image that the user can also preview on the right side. One design to note here is that only *Song Title* and *Artist* are mandatory fields, and they can not be modified after

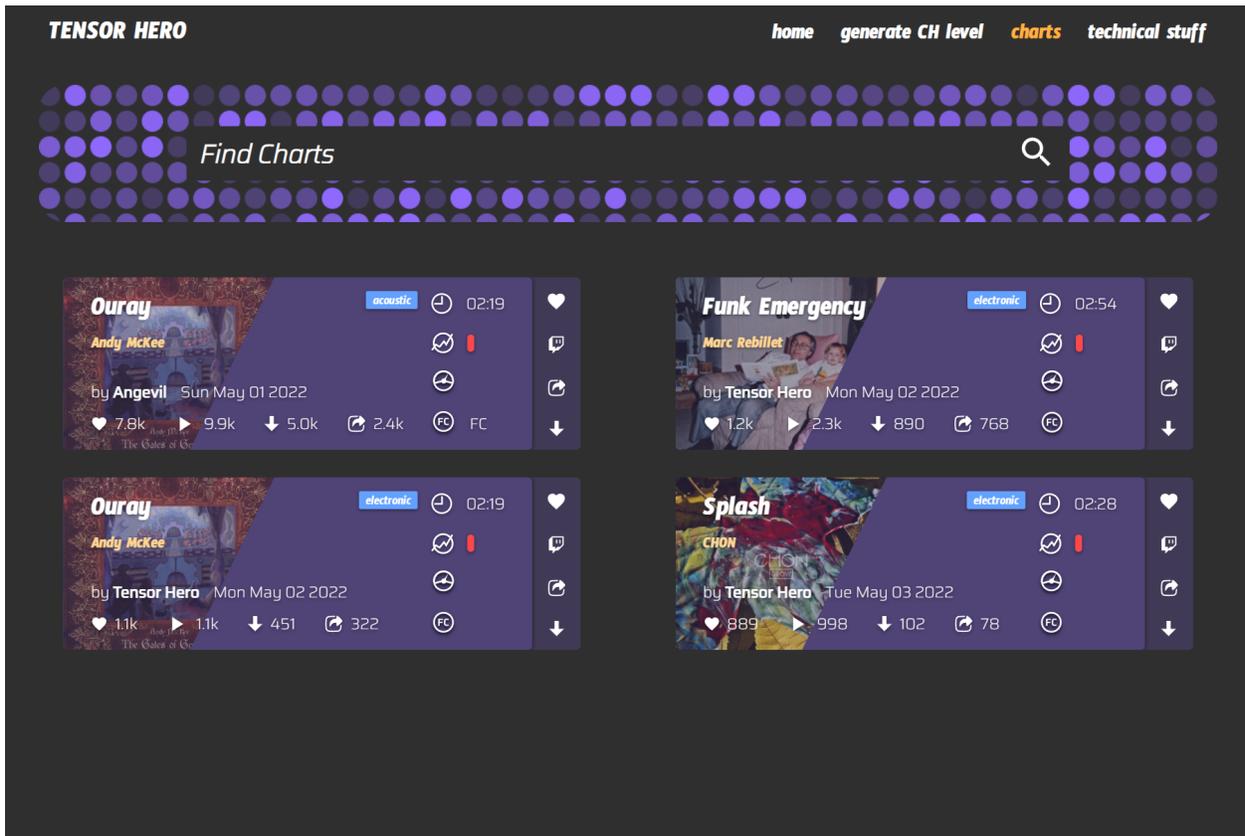
the chart is generated to prevent costly calls to the backend. However, the other fields are editable throughout the process, up until the user downloads the file package.



6.6 Chart Visualization

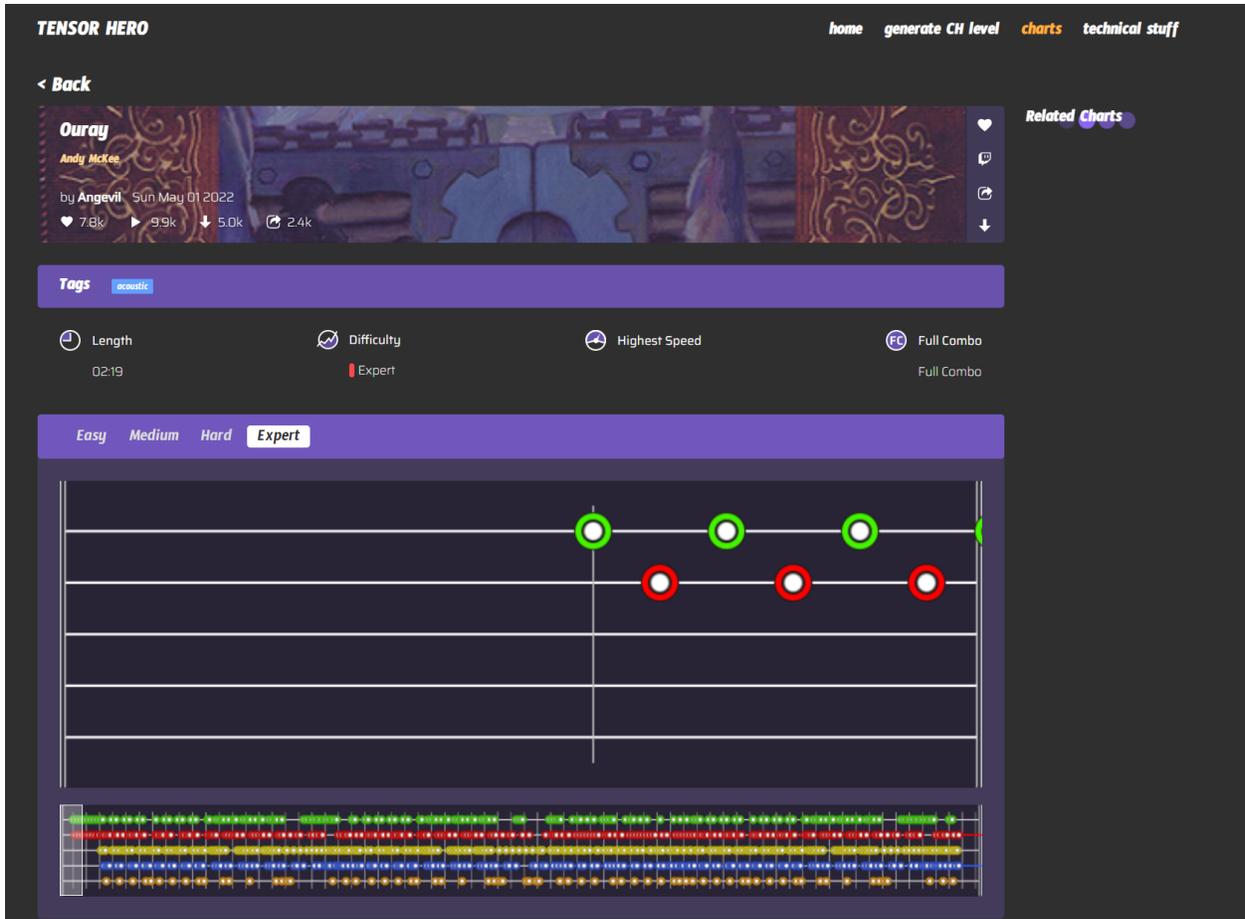
After uploading the music file, the model is put to work. While waiting for the chart to be generated, animations will help the users to understand how the machine learning model works, and the progress bar will show how far it is until the chart is generated.

After the whole process concludes, the user is taken to this chart visualization page. Here they can pick the different difficulty levels of the song, play the chart along with the music, and fill out the song information if they didn't do it earlier. If they're happy with the result, they can click on the download button, download the file folder that can be directly imported into Clone Hero, and play this chart pack.



6.7 Charts Hub

This is the hub for all charts uploaded to or generated by Tensor Hero. Users are able to search by artist and song title (more search and filter functionalities will be enabled in the future), and each song is presented in a card form that groups some useful information and features together for the users to interact with to understand on an abstract level, and decide whether they want to click on it and further explore the song.



6.8 Chart Details

Once the user decides to look into the chart or the song, they're directed to this chart details page, where they are able to see more information about it, as well as playing the corresponding chart visualization.

On the right side of the page, Tensor Hero will also make suggestions on the charts users might like based on the artist, playlist, or genre, which is a more advanced feature we plan to implement in the future.

What is Tensor Hero?

Tensor Hero is a web-based [Machine-Learning empowered](#) charting tool that auto-generates playable charts for Clone Hero (one sentence abt the community and support chart-sharing).

How It All Works

Curious what's happening under the hood? Below we explain how a song goes from an audio file to a chart file.

Step 1 SEPARATION

After uploading, the song audio file is immediately run through a source separation function, which allows us to extract only the guitar audio from the full song.

Step 2 SPECTROGRAMS

With just the guitar's audio, we convert the frequencies into a spectrogram, which is a standard data structure for analysis and processing of frequencies. You can think about it as a graph where the x-axis represents the timestep of the song, and the y-axis represents the observed frequency values.

Step 3 FLATTEN

We then take the spectrogram and squash it into a single data array, which is a fancy way of saying a list of values. These values represent, in order, the notes heard from the guitar.

Step 4 FEED

Now we are ready to feed this into the machine learning model. We cannot feed the whole song at once, so we break each song into pieces that are each about 4 seconds long, and feed them in order into the model.

Step 5 FLATTEN

As each piece is ingested, the model predicts corresponding CloneHero notes as output, which are then concatenated all together to make it a single long list of notes for the whole song.

Step 6 CHART-IFY

This output format, which is another long list of values, is then transformed into a chart format that is readable by the CloneHero game.

The TensorHero Model

The model that is predicting the notes for the chart file is a **Transformer**, a machine learning model that has been trained. We taught our model how to predict the notes by showing it hundreds of hours of examples of guitar audio with corresponding charts, both from the official GuitarHero game and CloneHero charters. The transformer model follows a neural-network architecture, and is essentially performing a type of "translation," where it's learning to equate one input "language" with another output "language." There are some great articles from the machine learning community exploring what Transformers are, how they work, and their applications in the real world.

More links How Transformers Work <https://towardsdatascience.com/transformers-141e32e69591>

More links What Is A Transformer? <https://blogs.nvidia.com/blog/2022/03/25/what-is-a-transformer-model/>

More links Transformers Illustrated <https://jalanmar.github.io/illustrated-transformer/>

6.9 How It All Works

One key element of our project is to provide transparency in regards to the ML model performing the chart generation. We decided to dedicate an entire page on our frontend and worked with the Tensor Hero Backend team to convey in manageable steps the neural network used in the chart generation. Initially we discussed providing a detailed description of the model, so users could understand the exact mechanisms behind the model. However, in one study conducted by Honeycutt et al., researchers found participants in fact had lower trust in the system and lower perception of system accuracy when learning about the mechanisms of the system (2020).

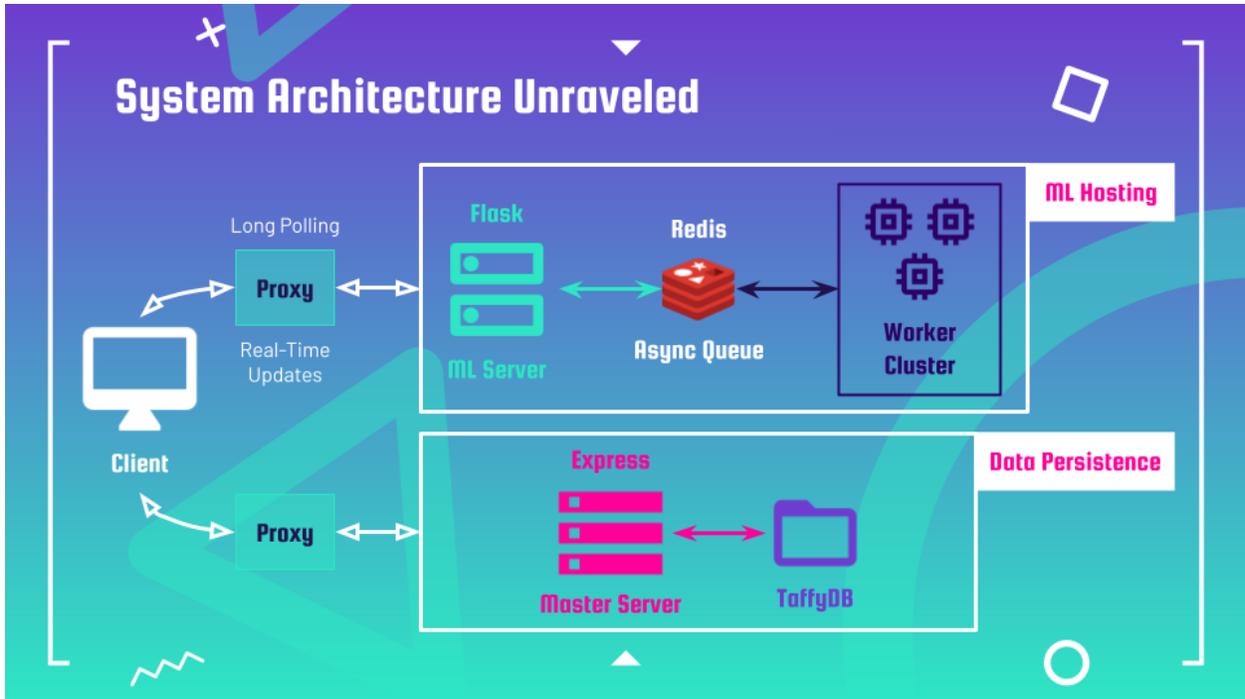
In the end, we decided to provide a high level overview, accomplished in six steps that describe the end-to-end process of audio ingestion to chart visualization. In addition to the text description, we created visualizations for each step that would serve as support and help with user understanding. For instance, the text description of spectrograms is difficult for someone unfamiliar with the concept without the visual as support. In our usability tests, subjects appreciated learning about the model and communicated that the high level description was indeed sufficient in understanding the overall process.

7. Implementation

The Tensor Hero website is developed using TypeScript and webpack. Source code is bundled and minified to reduce loading time and serve users a better experience.

7.1 System Architecture

Tensor Hero consists of three main components that work together to deliver a smooth end-user experience: the frontend, the master server, and the ML server. The frontend is the website users visit with their browser. The master server and the ML server are two isolated and standalone services that the frontend communicates with. The master server ensures data persistence so that charts published to the website can be seen by other users. The ML server handles the job of converting song audio to playable Clone Hero charts.



7.2 Real-Time Progress of the ML Model

The real-time progress of the ML model users will see after they upload their songs is achieved by long polling. When users click on the “submit” button in the upload form, a PUT request will be sent from users’ browser (the frontend) to the ML server. Upon receiving this request, the ML server will create a job, which is pushed into a Redis queue, for the ML server to process the song, and generate a *jobId* as the unique identifier of the job, which is sent back to the users’ browser as the response. The frontend will then schedule an interval event that asks the ML server every 2 seconds for the current progress of the ML model with this *jobId* through a GET request. This process is called *long polling*, which allows the frontend to update the model’s progress in real time.

The real-time progress of the ML model is one of the core features that demystify the ML model. It normally takes the ML model the amount of time comparable to the length of the song to generate the chart output. During this period, users will be able to see what the ML model is working on on the website in the form of animations and get a sense of how it converts the audio to the playable chart step by step.

7.3 Chart Visualization

The interface for chart visualization resembles a video player with an additional overview of the entire chart that can be used to navigate through the chart visualization. The chart generated by the ML model is in text format and will be parsed by the frontend to create the visualization. The fretboard (the background), the measure bars (the vertical lines), and the notes are all rendered inside the canvas element on the web page. The visualization itself follows the Clone Hero game. The song will be played in sync with the visualization so that users can feel how well the notes match the melody and rhythm of the song.

The main purpose of the chart visualization is to provide users with a direct way to evaluate the ML-generated charts. Otherwise, users will need to download the generated charts from the website first, upload them to Clone Hero or charting software, and then play the charts there. This back-and-forth workflow is inconvenient and counters our goal of making the charting process easier. Another benefit of offering the chart visualization is that it enables users to navigate through the chart. If they play the chart in the game, they have to play it from the beginning to the end. Although they can pause the game to see a certain section of the chart, they can never jump around and zoom in/out on the chart.

7.4 Downloadable Charts

Users are able to download a zipped bundle of the chart, the song, and the album cover directly from the website by clicking on the “download” button/icon. The bundle is structured so that Clone Hero can recognize and understand it, which reduces the users’ pain of adjusting in-game settings and configurations.

7.5 Open Source and Maintainability

The codebase of the Tensor Hero website is open to the public on GitHub. We imagine that Tensor Hero can become something that is widely shared across the Clone Hero

community and become a community-owned in-house artifact in the near future. To help realize this promise, it is important to ensure that all lines of code are readable and all functionality of the website is testable. Test-driven development and continuous integration are applied during the development of the site to ensure all builds and test runs are automated. Building and testing pipelines are set up using GitHub Actions, which are triggered on every push and pull request to the repository. All modules and UI components are also well-documented in the comment section of the code. In this way, developers from the community can easily follow the codebase and contribute to the project without accidentally breaking the existing functionality.

8. Findings from User Testing

Following deployment of a prototype site, we conducted several testing sessions using the platform UserTesting.com. This platform enables businesses to recruit testing audiences across different demographic and geographic profiles; testers, who are practiced at performing think-aloud tasks, agree to screen capture and audio recording of their sessions. Our full test script is included as an appendix – we tested each component of the site, both in presentation of information and activities supported. While our participants were not current Clone Hero players, we did screen for familiarity with it or other guitar-based rhythm games, so that they could provide both general (UI design) and contextualized feedback.

We received a lot of constructive and actionable feedback from our testers, including:

- **Functionality:**
 - Add selector for which difficulty level(s) to generate to Upload page (once supported by ML model)
 - Organize the chart library so that it can be searched by any metadata field (song title, album, year, etc.); filtered/sorted by song duration, difficulty level(s), genre

- Expand to additional audio file formats – currently .mp3, .wav, .ogg; could add .m4v and .flac support
- Change the default width of the adjustable box in the chart visualization tool; provide a clearer handle to indicate it can be manually resized
- Front-end / Visual Design:
 - Buttons on homepage, generate-chart page could convey stronger CTA
 - Chart library cards include some text superimposed over album cover image (which is uploaded by the user), may be difficult to read in some cases
 - Two-column cards in chart library have odd spacing between them in some aspect ratios, and are fairly information-dense in others
 - The menu bar for the chart visualization tool only appears when the cursor is nearby, which can lead to frustration
 - On chart page, could add more distinguishability between interactive (buttons) and non-interactive (headings, display icons) elements
 - Functionality icons on Charts pages lack hover-over text; the sets of icons along the right and bottom edges are inconsistent in both contents and ordering
 - External links on technical explanation page could be grouped together under a single header rather than each having its own
- Technical explanation content
 - Include a brief description of the project team and our goals for the site
 - Provide a high-level feature roadmap
 - Even out the level of technical detail in the step-by-step explanation – several are written with a general user in mind, while a couple are more technical and don't contribute to lay understanding of the process

9. Future design opportunities

We have a lot of features in mind to implement in the future from both the design and engineering perspective that's out of scope for capstone. For instance, "Account" feature is

essential in enabling high level user interactions and a sense of community, and by allowing the users to create their own accounts for the website, they can truly own their work and create meaningful relationships as part of the community network. We'd also like to establish "Tensor Hero" or an equivalent identity as a community member that represents the machine learning generated aspect of the website, to see how open and accepting people are towards this concept and how users interact with it. We'd also like to potentially allow for editability of the chart visualization and let players tailor the chart to their like, or as a way to help us train the backend model.

Works Cited

Honeycutt, Donald R., Mahsan Nourani, and Eric D. Ragan. 2020. "Soliciting Human-in-the-Loop User Feedback for Interactive Machine Learning Reduces User Trust and Impressions of Model Accuracy." *Proceedings of the Eighth AAAI Conference on Human Computation and Crowdsourcing*.
<https://arxiv.org/abs/2008.12735>.

Appendix X. Usability Testing Script

Screener

- Please select all from the following game franchises that you've played or heard of before?
 - Guitar Hero [May select]
 - Rock Band [May select]
 - Clone Hero [May select]
 - Frets on Fire [May select]
 - Not sure [Reject]
 - None of the above [Reject]
- Are you in a location where you can play audio? This could be a quiet place at home, or in a public place with headphones or earbuds.
 - Yes [Accept]
 - No [Reject]

Introduction

Clone Hero is a guitar-based rhythm game that is playable for free on a PC. Unlike previous games of the same type, players are free to make new levels using their own music and play them, or even publish them online. Currently, turning a song into a Clone Hero level (also called a "chart") is done by hand; some players now chart new songs as a hobby.

This scenario is a test of a website. There are no right or wrong answers for any of the questions -- you are not being tested, the software is.

Tasks

1. Without leaving the homepage, what is your first impression of this website? What do you think you will be able to do using the site? Please be specific.
2. Assume that you have a song that you want to turn into a chart using the site's Artificial Intelligence (AI) system. Find the page where you can upload a file. Do the text boxes you need to fill in on this page make sense to you? Do you think you'd be able to complete the form successfully? Explain your answer.
3. Launch URL:
<https://rexcheng1997.github.io/tensorhero-frontend/public/charts.html?chartId=0>
This is the screen you will see after the Clone Hero chart has been created. Take a minute or two to explore this page (if you are in a public place, note that the

interactive element plays audio when you start it). When you are finished exploring, move to the next task.

4. Stay on the page for this chart for "Ouray." This chart is an example, created by the player Angevil. Play the audio/video widget for a few seconds; skip to 1:10 into the song and play it for a few seconds more. Do the notes on the chart seem to match the music you hear? Why or why not?
5. Stay on the page for this chart for "Ouray." Looking at all the icons and words used on the page, which ones are hard to understand? Be as descriptive as possible, and explain what you think they might do.
6. Launch URL:
<https://rexcheng1997.github.io/tensorhero-frontend/public/charts.html?chartId=3>
This is a chart for a different song. As before, play a few seconds of the audio/video widget, then skip ahead to 1:30 in the song and play a few more. Do you notice any differences between this chart and the previous one? Please be specific.
7. Stay on the page for this chart for "Splash." This chart was created by an AI System. If you could provide feedback to the AI System about the chart, what would you tell it? Please be specific.
8. Navigate to the page that lists all of the charts that have been uploaded to the site. What is your first reaction to this page? Please explain your answer.
9. Stay on the page that lists all of the charts. Besides "song title," what other ways would you like to be able to search for music on this page?
10. Launch URL:
<https://rexcheng1997.github.io/tensorhero-frontend/public/about.html>
This page summarizes what the AI System does to make a chart when you upload a song. Spend a minute or two reading through this explanation. On a scale of 0 (none of it makes sense) to 10 (I understood perfectly), how would you rate this page? Which part is most difficult to understand?
11. Stay on the "technical stuff" page. Now that you've read this explanation, what questions do you have for the team that built this site?

Overall Questions (text response)

1. What frustrated you most about this site?
2. If you had a magic wand, how would you improve this site?
3. What did you like about the site?
4. How interested are you now in trying the game Clone Hero (0=Not at all interested, and 10=Very interested)?