

Worcester Polytechnic Institute

Digital WPI

Interactive Qualifying Projects (All Years)

Interactive Qualifying Projects

2020-06-09

Jazz History Database Global Contributor Project

Lucas Varella

Worcester Polytechnic Institute

Mikel Matticoli

Worcester Polytechnic Institute

Follow this and additional works at: <https://digitalcommons.wpi.edu/iqp-all>

Repository Citation

Varella, L., & Matticoli, M. (2020). *Jazz History Database Global Contributor Project*. Retrieved from <https://digitalcommons.wpi.edu/iqp-all/5826>

This Unrestricted is brought to you for free and open access by the Interactive Qualifying Projects at Digital WPI. It has been accepted for inclusion in Interactive Qualifying Projects (All Years) by an authorized administrator of Digital WPI. For more information, please contact digitalwpi@wpi.edu.

Development of the Jazz History Database Global Contributor System



By:

Mikel Matticoli Computer Science + Interactive Media & Game Dev
Lucas Varela Computer Science



Development of the Jazz History Database Global Contributor System

An Interactive Qualifying Project
Submitted to the Faculty of
WORCESTER POLYTECHNIC INSTITUTE
In partial fulfillment of the requirements for the
Degree of Bachelor of Science

by
Mikel Matticoli
Lucas Varella

Date:
8 June 2020

Report Submitted to:

Professor Richard Falco
Key Stakeholder
Jazz History Database

Professor Keith Zizza
Project Advisor
Worcester Polytechnic Institute

Abstract

The JazzHistoryDatabase is a non-profit organization at WPI that archives recordings, photographs, and other jazz artifacts from around the world that might otherwise deteriorate. This archive, accessible at jazzhistorydatabase.com, was previously maintained by students and faculty who built the website by hand. Our goal was to design, build and document a web-based tool to allow volunteers and international correspondents JHDB to upload digitized jazz artifacts into a simple web form that would subsequently output and publish template-based web pages.

Table of Contents

Abstract	2
Table of Contents	3
Acknowledgements	6
Executive Summary	7
Background and Introduction	7
Importance	7
Target Audience Evaluation	8
Objectives Achieved	9
Deliverables	11
Contributor Portal	11
Tutorial Video	11
Trailer Video/Virtual Research Showcase	11
Technical Documentation	11
Conclusions & Recommendations	11
Scope is Key	12
Focus on Users	12
Future Work	12
Authorship	13
List of Figures	14
Introduction	16
Background	18
What is the Jazz History Database?	18
Importance	18
Reach	19
Day to Day Operations	20
The Problem	21
Approach: Why Start From Scratch?	22
Previous Work	22
Methodology	23
Objectives	23
Target Audience Evaluation	24
UI/UX Design	26
Development Workflow	28
Technologies Used and Supporting Systems	29
Production Environment/Infrastructure	29

Technical Implementation	32
Component Hierarchy (User Interface Breakdown)	32
Header	33
Information Tab	35
Media Upload Tab	36
Pages Tab	37
Edit Tab	38
MediaUpload Component	40
FileUpload Component	43
Preview Tab	45
Evaluating User Experience	46
Study Objective	46
Research Methods	46
Risks	47
Study Methodology	47
Sample Survey Questions	47
Obstacles	48
Scope	48
Quaran-term	48
Technical Constraints	49
Outcomes	50
Objectives Achieved	50
Deliverables	55
Contributor Portal	55
Tutorial Video	55
Trailer Video/Virtual Research Showcase	56
Technical Documentation	56
Conclusions & Recommendations	57
Scope is Key	57
Focus on Users	57
Future Work	58
References	59
Appendix A: Sample User-Study Invitation	61
Email Invitation	62
Appendix B: Survey Questions	64
Jazz History Database Contributor Portal Survey	64
Appendix C: Survey Results	69
Appendix D: On-Boarding Documentation	73

Acknowledgements

This project was proposed in collaboration with the Jazz History Database (JHDB) at Worcester Polytechnic Institute. The accomplishments detailed in this paper would not be possible without the efforts and insight of talented and passionate individuals from many different backgrounds. Our team would like to thank Professor Rich Falco, the JHDB's founder and the driving force behind the important services it provides, as well as Professor Mike Drnek, the mastermind behind much of the original technology powering the JHDB website and a critical source of information and insight into the JHDB's operations and legacy infrastructure. Without their passion and guidance, this project never would have started, let alone made it through the early research and design phases and expanded into an IQP.

We would like to thank Professor Keith Zizza of the Interactive Media & Game Development Department, our IQP advisor, for his passion, enthusiasm, and commitment to this project in spite of less than ideal circumstances. As a musician, composer, and audio expert with a technical background, Professor Zizza joined the project at the perfect time, and enabled us to focus our scope and develop the most mission-critical components of this project.

We would additionally like to express our gratitude to Professor Charlie Roberts of the Computer Science and Interactive Media & Game Development Departments, who advised the prototyping phase of the project in the form of an independent study in D term 2019. Professor Roberts provided insight into a range of possible solutions we had not considered, as well as expertise on the technologies we ultimately chose and informed several early design decisions both directly and indirectly.

Lastly, we would also like to thank Tom Bellino, the JHDB's Global Correspondent, as well as the student builders at the JHDB who provided critical feedback on early prototypes of the tool that helped to inform the design and functionality of the final product.

Executive Summary

Background and Introduction

The Jazz History Database (JHDB) is a non-profit organization at Worcester Polytechnic Institute (WPI) that archives recordings, photographs, and other jazz artifacts from around the world that might otherwise disappear over time. For every big name in jazz, there are dozens of great artists who don't make it to the national scene. The goal of JHDB is to celebrate the accomplishments of these artists, whether in music, poetry, visual art, or any other field. Decades of work have gone into preserving countless jazz artifacts since the JHDB was founded in 1999. This archive is publicly accessible via the JHDB website (jazzhistorydatabase.com), currently maintained by students and Professors Richard Falco and Mike Drnek at WPI.

The website routinely accepts contributions from external reputable sources (archivists, artists, composers, etc.). Extensive outreach efforts by Professor Falco brought the first international correspondents to contact the JHDB in pursuit of contributing to the archive. This development possessed incredible potential. The JHDB now had the chance to not only continue to be an online museum for local and U.S.-based jazz history, but to fulfill its goal for the entire global jazz community. However, this would also lead to an unprecedented increase in responsibilities for the students and staff. Even before the organization “went global” with correspondents, the team already found itself behind schedule, with several tasks piling up in their backlog.

We identified two core problems with the page-building workflow that were creating severe bottlenecks in the publishing process: large amounts of unnecessarily repeated manual work for small pages, and inconsistencies introduced by human error across the output of this repeated work. It is reasonable to expect that students with different levels of technical and design expertise would produce different results while trying to create similar content. By creating a system that takes in the media and feeds it into a reusable template, the repeated aspects of the process are removed and outputted pages with the same structure are uniform, and not subject to formatting errors.

Additionally, much of this media was already being submitted in a digital format by contributors, which posed the question: if some contributors were already able to upload this media to the JHDB Dropbox, and students are simply inputting the media into a web form to generate the pages, why not give contributors access to create the pages themselves? As such, our third problem became how to enable contributors with pre-digitized content to create pages themselves.

We hope this tool will allow developers and correspondents to seamlessly contribute to the site, that the JHDB will thrive and use it to mitigate their backlog, and that it will help the international jazz community to preserve and celebrate their history.

Importance

The reach of the Jazz History Database extends far beyond New England. Google Analytics results for jazzhistorydatabase.com show that, in March 2020 alone, the website was visited by users in 58 different countries across 6 continents. Since 2010, jazzhistorydatabase.com has been visited by nearly a hundred thousand unique users from 178 countries. While these statistics stand for themselves, it is

worth noting that the JHDB serves a very niche audience, and to date has not had the resources to localize the majority of its content in any language other than American English.

Some content submitted to the JHDB for archival exists only in very antiquated formats that are becoming increasingly difficult to find. While it is already difficult to digitize media in recently retired formats such as VHS tapes, a significant portion of the media is received in even older forms: vinyl records, phonograph cylinders, paper manuscripts and hand-written sheet music with the original crossouts. Without the work of the JHDB, locally famous jazz artists who produced this content may be forgotten, and the artifacts themselves lost to effects of time on these physical mediums. The JHDB additionally collects biographies and hosts interviews with many locally famous artists, whose stories and work would otherwise remain undocumented and eventually be lost. Professor Falco has personally interviewed numerous such artists and their families.

With both Professor Falco and Professor Drnek retiring, and major funding cuts due to COVID-19, it became critical to the survival of the JHDB to reduce the time, resources, and technical skill required to publish the JHDB's extensive backlog of digitized content. Our work has allowed non-technical volunteers of the JHDB, including Professor Falco himself, to quickly build and publish tens of pages that would otherwise have taken the span of weeks or months to get published with limited staffing and funding.

Target Audience Evaluation

To identify the needs of our audience, we employed a purely qualitative approach. Since our audience was primarily non-technical, we concluded it would be easier for them to communicate their concerns in a more anecdotal/conversational format, as opposed to answering predefined questions on their expectations of the functionality. From these conversations, we identified a few key focus points:

- **Phrasing** - our target audience consisted largely of jazz musicians and historians without technical backgrounds. As such, clear and unambiguous phrasing of instructions and button/form labels were critical. Simple details such as referring to built collections as “contributions” could be misconstrued to imply monetary contributions.
- **Layout** - we initially planned to base our design on Google's Material Design specification (Material design). While most of the design principles such as visual consistency and responsiveness carried well, we opted to stray from the guidelines in certain key areas. Most notably, icons that have a clear and consistent meaning for frequent users of modern websites/apps, such as the hamburger menu icon, did not effectively communicate the associated functionality for non-technical users. This was addressed by clearly labeling buttons with their functionality to avoid any possible confusion/ambiguity.
- **Colors** - by default, the UI component framework we used consists of a light theme with bright, warm colors to denote functional UI elements (navigation bars, buttons). While there is nothing inherently wrong with these color choices from a technical standpoint, we found that they did not give off “jazz vibes.” We opted to change this in favor of the dark theme to reduce eye strain along with light blue highlights to match the color palette of jazzhistorydatabase.com. We received immediate positive feedback on this change, as the cooler, less harsh color scheme fit with predefined associations that some users had with the colors (light blue equals jazz) and provided an overall “clean” aesthetic.

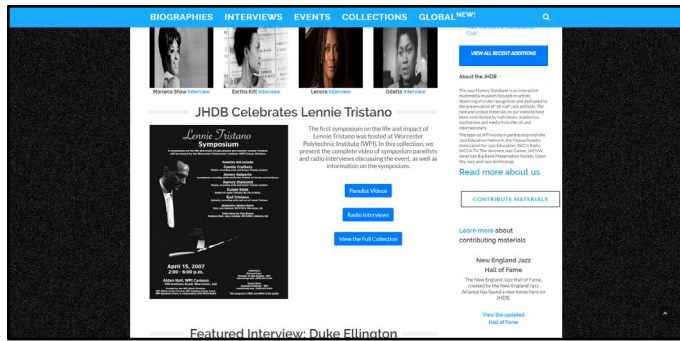


Figure 1. Light blue color palette with dark grey background at jazzhistorydatabase.com.

Objectives Achieved

- The contributor portal [it] must be available online at all times.*

Thanks to careful research on and implementation of current technologies and Site Reliability Engineering standards as detailed in our methodology, the contributor portal is available globally with near-100% uptime. Due to the inherent possibility of unpredictable outages with network-based technologies, we successfully laid out countermeasures to allow us to quickly detect and resolve outages.
- It must require little familiarity with computers and minimal technical expertise.*

While we were unable to survey actual contributors due to COVID-19 related complications, our study group was overall a strong representation of our target audience, with subjects that had some familiarity with computers and varying degrees of familiarity with the JHDB's operations

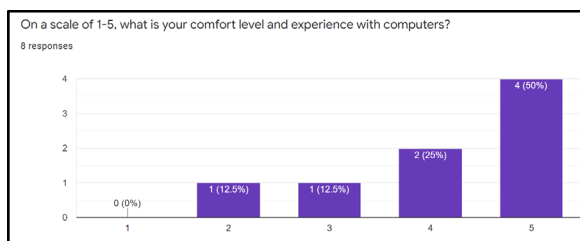


Figure 2.1. Subjects ranged from low-moderate to high levels of comfort with computers, with the majority feeling relatively comfortable. This is a reasonable audience to assess as we are assuming most users of the portal have first-hand or second-hand familiarity with digitizing artifacts.

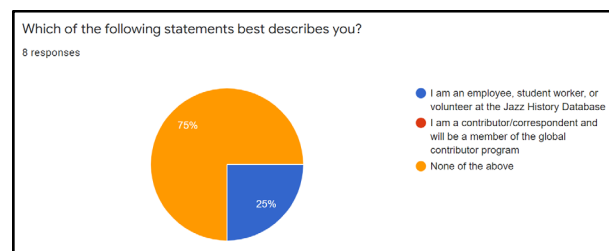


Figure 2.2. 75% of subjects were generally unfamiliar with the day-to-day operations of the JHDB, with 25% being JHDB workers.

- It must feature various sources of aid and clarification to the user.*

We were successfully able to develop a professional and thorough video tutorial on using the contributor portal. While we did not directly survey users on the quality of the video, several subjects expressed in their comments that they found the video helpful and we did not receive any negative feedback on the video.
- It must allow the user to upload text, image, audio, and video media into pre-build web page templates.*

This functionality was achieved in full, and users are able to upload images and audio as well as paste text into the portal. All subjects successfully created a page and linked media to it, though

one subject expressed that they were unsure whether that step worked due to a malfunction that prevented them from previewing the page.

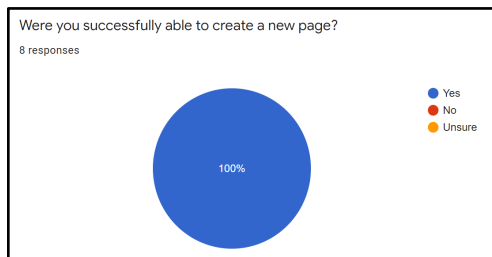


Figure 3.1. Pie chart of users able to create a page.

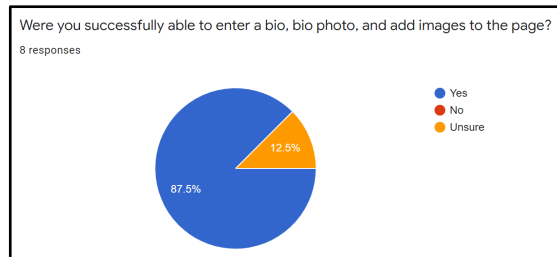


Figure 3.2. Pie chart of users able to enter bio, bio photo, and images to page.

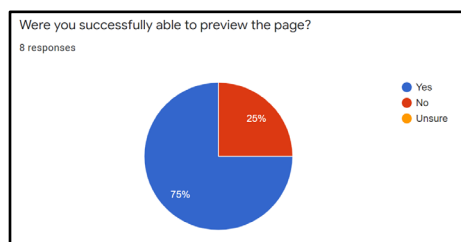


Figure 3.3. Pie chart of users able to preview the page.

2/8 subjects were unable to preview the page to see what it looked like. We were successfully able to identify the underlying bug in the preview code, and resolve it so that it will not pose an issue in the future ("! PROD RENDERER 502," 2020).

Of the subjects that were successfully able to view the preview, all of them expressed that the resulting preview felt representative of the information they entered into the portal.

- *It must allow the user to request approval to publish a collection of web pages. Once approval is electronically granted by an administrator (such as Professor Falco), the collection shall be published at global.jazzhistorydatabase.com.*

Upon completion of the survey, all user-created pages were marked as "Review Pending," indicating that subjects had successfully requested review. Professor Falco was successfully able to make revisions and subsequently publish these pages.

- *It must allow the user to create and maintain multiple collections.*

This functionality was inherent in our design. Through his own usage after the v1 release, Professor Falco was able to quickly and easily produce and publish several pages, now live at <https://global.jazzhistorydatabase.com>

- *It must sandbox users to their own particular 'branch' of the global JHDB, such that a correspondent will not have access to another correspondent's collections, unless they express that they work on the same collections.*

This functionality was achieved through creating multiple levels of access. Contributor-level accounts only have access to their own pages and files, while administrators are able to view all pages in order to assist the JHDB's correspondents and publish pages.

- *Its source code and repositories must include extensive documentation, to facilitate onboarding future developers into the project.*

The code for the contributor portal is open source, and is publicly available on GitHub at <http://github.com/jazzhistorydatabase/jhdb-cms>. The on-boarding process is thoroughly documented in a README file in accordance with industry standard practices. A copy of the README file is included in Appendix D.

- *Lastly, the portal must protect the integrity and identity of the JHDB and its members, such that no personally identifiable information is unintentionally publicized, and no remote correspondent could accidentally damage the portal or the website.*

By employing industry best practices for security and ensuring that the portal did not store any identifying information for contributors in the first place, we were successful in maximizing both ease of use for our users as well as data security and user privacy.

Deliverables

Contributor Portal

Our primary deliverable was the contributor portal itself, which is online at <https://contributor.jazzhistorydatabase.com>. If anyone reading this paper would like to learn more about the JHDB and the contributor portal, they can learn more at <https://global.jazzhistorydatabase.com> or reach out to the team directly via email at global@jazzhistorydatabase.com.

Tutorial Video

The instructional video explaining the use of the portal was our second key deliverable. This video is public at <https://contributor.jazzhistorydatabase.com/> such that prospective contributors can view the video and familiarize themselves with the portal before even receiving their login credentials. The video runs about 15 minutes, and provides a well-paced step-by-step walkthrough starting with obtaining their login credentials, logging into the portal, all the way through building and publishing their first page.

Trailer Video/Virtual Research Showcase

Since International Jazz Day fell within the timeframe of our IQP, we also created a general informational video on the JHDB and the Global Contributor Project. This trailer video is posted on the frontpage of <https://global.jazzhistorydatabase.com>. A variant of this video and an accompanying Powerpoint presentation were submitted for WPI's [Virtual Research Showcase](#), which included additional background and historical context for the IQP and project specific to a WPI audience.

Technical Documentation

Our last deliverable was technical and procedural documentation of our process. This documentation was developed with the goal of being sufficient to hand the project off entirely to someone with the required technical knowledge to continue to maintain it in our stead. The procedural parts of this documentation were provided in the form of the aforementioned videos and this paper. Technical documentation is provided alongside the code, and a copy of the on-boarding documentation for development of the portal is included in Appendix D.

Conclusions & Recommendations

The core project was completed in eight weeks - an extremely ambitious timeline for a project with such a vast scope. While our team began the project with working knowledge of the JHDB and

technologies used, the process was not without its hiccups and we provide our concise reflections below in hopes that they will prove useful to future project teams. (To anyone pursuing a project similar in scope, we recommend reviewing the full [Conclusions & Recommendations](#) section).

Scope is Key

It is critical to understand the needs of the project as thoroughly as possible, and to identify the highest value action items at any given time. This practice is not limited to the planning phase of the project - constantly and repeatedly evaluating, prioritizing, and down-scoping is always necessary to deliver a usable product by a deadline. For others pursuing this kind of project in the future, we highly recommend Jeff Sutherland's book *Scrum: The Art of Doing Twice the Work in Half the Time* (2016).

Focus on Users

Understanding the needs of stakeholders and addressing their ideas is important, with the context that they may not have a working knowledge of the technical constraints at play or an awareness of some viable solutions.

It is key to ask a lot of questions and build an understanding of the core problem facing the users of your tool. The solution may not line up one-to-one with their suggestions, but it is important to understand the context you may be lacking that informs those suggestions. Most importantly, always remember that your tool will be used by a human being, with different background, experiences, and expectations from you. A step or detail that seems obvious to you may be thoroughly confusing to your tool's target user. Make sure to deliver tangible prototypes early that test your assumptions about the user's expectations, and ask questions to contextualize the user's feedback on what you show them. Focusing on the end-user of a product is always critical to developing a tool that will be not only functional, but intuitive and even enjoyable.

Future Work

While the scope of this project is completed, there is a lot of room for future work on the contributor portal. There are various minor bugs and potential optimizations that need to be addressed, as well as additional use cases for the portal to be expanded to encompass (adding more templates, or even a template creator into the portal, linking pages together, etc). On our part, we took on this project because we recognized the unique value of the JHDB and the critical need for this technology in allowing the JHDB's services to remain available to the global jazz community. Our development work on the project is ongoing, and open-source contributors (both from fellow WPI students/faculty and external assistance) are greatly welcome.

Authorship

Authorship of this document was equally shared by Mikel Matticoli and Lucas Varella, with feedback from our Advisor Keith Zizza. Portions of each section were written and revised multiple times by both authors, and all citations, list of figures, and appendices were also compiled in collaboration after all revisions were made.

Mikel was primarily responsible for the Background section, and user experience and technical documentation portions of the methodology. Lucas was primarily responsible for the Introduction section, as well as the user interface breakdown and workflow portions of the methodology. Objectives were initially defined by Lucas, with feedback from Mikel and Professor Zizza, and subsequently addressed in the outcomes section by Mikel. The conclusions and recommendations section was written by both Lucas and Mikel through a thorough retrospective discussion over video conference. Mikel was primarily responsible for compiling citations and converting references to APA format, as well as the on-boarding documentation in Appendix D. Lucas was primarily responsible for compiling the other appendices, as well as the list of figures. The executive summary was initially compiled by Mikel, and subsequently trimmed down to a reasonable length by Lucas. Finally, the abstract and authorship sections were authored painstakingly in tandem over video conference, as it is very difficult to summarize an eight-week project with a one-year lead time in eighty words.

List of Figures

- Figure 1. Light blue color palette with dark grey background at jazzhistorydatabase.com.
- Figure 2.1. Subjects ranged from low-moderate to high levels of comfort with computers.
- Figure 2.2. 75% of subjects were generally unfamiliar with day-to-day operations of the JHDB.
- Figure 3.1. Pie chart of users able to create a page.
- Figure 3.2. Pie chart of users able to enter bio, bio photo, and images to page.
- Figure 3.3. Pie chart of users able to preview the page.
- Figure 4.1. Screenshot of Steve Cancelli Collection landing page on jazzhistorydatabase.com.
- Figure 4.2. “Re” by Steve Cancelli - an original manuscript from the JHDB.
- Figure 4.3. “Steve Cancelli Interview with Rich Falco”.
- Figure 5.1. New users by country world map (Mar 1, 2020 - Mar 31, 2020).
- Figure 5.2. New users by country, top 10 countries (Jan 1, 2010 - May 1, 2020).
- Figure 5.3. New users by country world map (Jan 1, 2010 - May 1, 2020).
- Figure 6. Standard Workflow for Jazz History Database Collections.
- Figure 7. File storage structure from 2016 attempt at a JHDB content management system.
- Figure 8.1. Old prototype for login flow.
- Figure 8.2. Updated login flow.
- Figure 9. Light blue color palette with dark grey background at jazzhistorydatabase.com.
- Figure 10. Global Contributor Portal landing page.
- Figure 11.1. Portal menu drawer.
- Figure 11.2. Portal notice that the user is being redirected for login.
- Figure 12. Contributor Portal Information Screen immediately after logging in.
- Figure 13.1. Amplify dashboard shows at-a-glance system health.
- Figure 13.2. Amplify real time usage statistics and analysis.
- Figure 13.3. System outages trigger immediate email alerts from Amplify.
- Figure 14. Component hierarchy of the portal’s client-side React.js code.
- Figure 15. Contributor Portal Header Component.
- Figure 16. The menu on the top left of the Header, as it appears when opened.
- Figure 17. The Help Dialog displayed once the Help button is clicked in the website menu.
- Figure 18. Menu displayed when the user has logged in with an administrator account.
- Figure 19. The landing page of the contributor portal, under the Information tab.
- Figure 20. The UploadView of the contributor portal, under the Media Upload tab.
- Figure 21. Dropbox upload site, as it looked at the time of this writing.
- Figure 22. The ContributionListView Component. No contribution has been selected.
- Figure 23. The ContributionListView Component. A contribution has been selected.
- Figure 24. The EditContributionView Component, under the Pages tab, and the Edit tab.
- Figure 25. The Biography section of the EditContributionView.
- Figure 26. The Biography section of Bill Warfield’s published page.
- Figure 27. The Images section of the EditContributionView for the Bill Warfield page.
- Figure 28. The Images section of Bill Warfield’s published page.
- Figure 29. Part of the Audio section of the EditContributionView for Bill Warfield’s contribution.
- Figure 30. Part of the Audio section of Bill Warfield’s published page.
- Figure 31. The Video section of the EditContributionView for an example contribution.
- Figure 32. The Videos section of an example page.
- Figure 33. An instance of the FileUpload component in the Bio section.
- Figure 34. An instance of the FileUpload component in Bill Warfield’s Bio section.

- Figure 35. A group of FileUpload components in the Images section.
- Figure 36. A group of FileUpload components in the Audio section.
- Figure 37. A group of FileUpload components in the Video section of an example contribution.
- Figure 38. The Dropbox Chooser API website at the time of writing.
- Figure 39. The Preview secondary tab under the Pages tab.
- Figure 40. Email sent to study subjects.
- Figure 41. Summary of participants' comfort level with computers.
- Figure 42. 75% of subjects were unfamiliar with JHDB's day-to-day operations (enlarged).
- Figure 43. Pie chart of users able to create a page (enlarged).
- Figure 44. Pie chart of users able to enter bio, bio photo, and images to page (enlarged).
- Figure 45. Pie chart of users able to preview the page (enlarged).
- Figure 46.1. Regular users are only able to see their own pages.
- Figure 46.2. Administrators see all pages, but still see regular user pages listed below their own.
- Figure 47. Firestore rule preventing non-administrator users from accessing other users' data.
- Figure 48. Sample logs for publish endpoint. User and data identifiers have been redacted.
- Figure 49. Active project board on the contributor portal GitHub Repository.
- Figure 50. Survey question 1.
- Figure 51. Survey question 2.
- Figure 52. Survey question 3.
- Figure 53. Survey question 4.
- Figure 54. Survey question 5.
- Figure 55. Survey question 6.
- Figure 56. Survey question 7.
- Figure 57. Survey question 8.
- Figure 58. Survey question 9.
- Figure 59. Survey question 10.
- Figure 60. Survey question 11.
- Figure 61. Survey question 12.
- Figure 62. Survey question 13.
- Figure 63. Survey question 14.
- Figure 64. Survey question 15.
- Figure 65. Summary of results for question 1.
- Figure 66. Summary of results for question 2.
- Figure 67. Summary of results for question 4.
- Figure 68. Summary of results for question 6.
- Figure 69. Summary of results for question 8.
- Figure 70. Summary of results for question 10.
- Figure 71. Summary of results for question 15.
- Figure 72. On-boarding documentation.

Introduction

The Jazz History Database (JHDB) is a non-profit organization at Worcester Polytechnic Institute (WPI) that archives recordings, photographs, and other jazz artifacts from around the world that might otherwise disappear over time. For every big name in jazz, there are dozens of great artists who don't make it to the national scene. The goal of JHDB is to celebrate the accomplishments of these artists, whether in music, poetry, visual art, or any other field. Decades of work have gone into preserving countless jazz artifacts since the JHDB was founded in 1999. This archive is publicly accessible via the JHDB website (jazzhistorydatabase.com), currently maintained by students and Professors Richard Falco and Mike Drnek at WPI.

The website routinely accepts contributions from external reputable sources (archivists, artists, composers, etc.). Extensive outreach efforts by Professor Falco brought the first international correspondents to contact the JHDB in pursuit of contributing to the archive. This development possessed incredible potential. The JHDB now had the chance to not only continue to be an online museum for local and U.S.-based jazz history, but to fulfill its goal for the entire global jazz community. However, this would also lead to an unprecedented increase in responsibilities for the students and staff. Even before the organization “went global” with correspondents, the team already found itself behind schedule, with several tasks piling up in their backlog. The website was making its first impression on the international community. The reputation of the JHDB was at stake.

At least two independent study projects at WPI attempted to mitigate their growing backlog. These projects focused not on backlog items directly, but rather on improving the team's workflow. In 2016, an independent study project first explored a way to expedite the website's publishing process for new pages. The proposed solution expected web developers to install software that creates a template they could use as a starting point. Aside from its premature conclusion, this solution failed to support different templates and use cases. It would expedite only the creation process for one of the many webpage categories the site hosts, only minimally improving the developers' workflow. Nonetheless, it planted the seeds for another independent study project, and another approach.

By the Spring of 2019, the JHDB gained visibility with the international jazz community. Archivists from Europe expressed interest in serving their content on the website. Unlike U.S.-based contributions, obtaining original artifacts from international contributors for preservation would be often infeasible and risked the safety of the artifacts. The JHDB could still aggregate and showcase these artifacts, assuming international archivists preserve artifacts themselves. Professor Falco began recruiting students for another independent study project, aimed at enabling correspondents and JHDB developers alike to directly contribute to the website, requiring minimal technical skill from correspondents. This effort drew the attention of two JHDB developers and a WPI student. The work-study developers, Veronica Gurnawan and Lucas Varella, brought their expertise and familiarity with the JHDB workflow to the team. The student, Mikel Matticoli, brought in industry experience and a fresh perspective. We all shared an interest in preservation, and a passion for using technology to solve everyday problems for the layperson.

We proposed the development of a separate web portal where users upload media to the JHDB site. The portal inserts the users' media into configurable templates, builds new web pages from the templates, and automatically publishes them to the site. This workflow allows correspondents to publish new content to the site without involvement of JHDB developers by design. It also allows developers

themselves to publish and modify standard pages on the site, aiding their ever-growing backlog. Goals and ambitions were high. Deliverables were achievable, but we lacked experience with productivity tools and workflow standards. Examples include inconsistent task board usage, inconsistent testing and code reviews, and manual integration and deployment (as opposed to a continuous integration and deployment solution). We also lacked plans to conduct formal user testing. Soon the scope of the independent study shrunk into creating a proof of concept to prove we chose the right technologies for the job. We achieved a working prototype by the end of the independent study project, but nothing close to a software release. However, it proposed a viable solution to JHDB's needs, and it laid the groundwork for this IQP.

Armed with critical knowledge of the successes and shortcomings of past projects, our team sought to develop and release the tool they needed. As word spread about the website among the international jazz community, with interested correspondents on the brink of frustration, the stakes for the JHDB's reputation were higher than ever. Our goal was to empower developers and remote correspondents of the JHDB with the ability to contribute to the JHDB website remotely, seamlessly, and with little technical skill requirements. To accomplish this goal, we built a web-based tool (a "contributor portal") where developers and remote correspondents may upload media into a simple web form, which outputs web pages from reusable templates and publishes them at <https://global.jazzhistorydatabase.com>. We announced the upcoming release of this tool on the JHDB website, and are fixing minor bugs prior to release. We created this tool using the 2019 project as a backbone, implementing new technologies and methodologies on top of it. Our team took a lesson from the past projects and maintained a steady workflow, executing user testing surveys to expose our tool to representative users and incorporate their feedback into it. We hope this tool will allow developers and correspondents to seamlessly contribute to the site, that the JHDB will thrive and use it to mitigate their backlog, and that it will help the international jazz community to preserve and celebrate their history.

Background

What is the Jazz History Database?

Originally established to preserve local jazz history culture in New England, the Jazz History Database (JHDB) is a non-profit organization based out of Worcester Polytechnic Institute (WPI), and a service of the Massachusetts Association for Jazz Education (Jazz history database). Its mission has since expanded, and the JHDB now receives jazz artifacts from contributors all around the world in various formats, from pre-digitized content to original photographs, newspaper clippings, paintings, tapes, and wax discs. Many of these artifacts are unique, and at high risk of deteriorating or being permanently lost. For non-digital media, trained students are responsible for converting the media to a digital format and editing or remastering it as needed. All of this digital content is subsequently archived, and some of it showcased in the JHDB online museum at jazzhistorydatabase.com with the permission of those contributing the material.

Importance

As mentioned above, the JHDB digitizes, preserves, and archives countless pieces of unique jazz media. Much of this content exists only in very antiquated formats that are becoming increasingly difficult to find. While it is already difficult to digitize media in recently retired formats such as VHS tapes, a significant portion of the media is received in even older forms: vinyl records, phonograph cylinders, paper manuscripts and hand-written sheet music with the original crossouts. Without the work of the JHDB, locally famous jazz artists who produced this content may be forgotten, and the artifacts themselves lost to effects of time on these physical mediums. The JHDB additionally collects biographies and hosts interviews with many locally famous artists, whose stories and work would otherwise remain undocumented and eventually be lost. Professor Falco has personally interviewed numerous such artists and their families.

One of many examples of this is the Steve Cancelli collection ("Steve Cancelli," n.d.). According to the JHDB, Cancelli was born in 1941 and had his first paid performance at a Worcester church at the age of 14. The JHDB features various audio and video recordings of his performances, interviews with him and his daughter, photos, and manuscripts in his own handwriting. His music was greatly appreciated within his own community, though he was never well known outside of the local Worcester area. In January 2014, he was interviewed at WPI by Rich Falco, the video of which is posted on the Videos page of his collection alongside some subsequent performances in February of that same year.

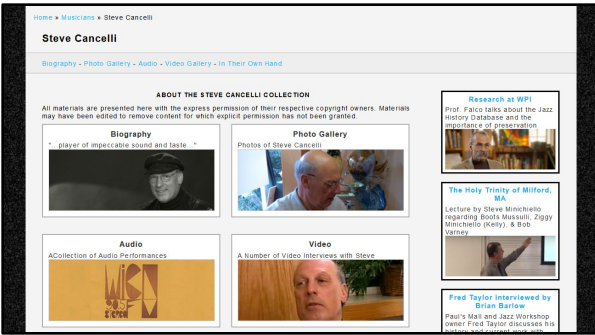


Figure 4.1. Screenshot of Steve Cancelli Collection landing page on jazzhistorydatabase.com. ("Steve Cancelli," n.d.)



Figure 4.3. "Steve Cancelli Interview with Rich Falco". ("Steve Cancelli - Photo Gallery," n.d.)

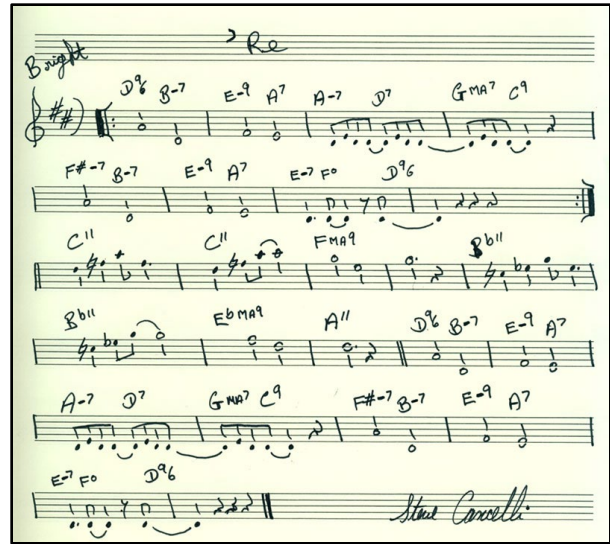


Figure 4.2. "Re" by Steve Cancelli - an original manuscript from the JHDB "In Their Own Hand" collection. ("Steve Cancelli," n.d.)

A Google search for Steve Cancelli yields only his biography on jazzhistorydatabase.com, and his obituary, dated November 24, 2016—just over two years after being spotlighted by the JHDB ("Steven Cancelli Obituary," 2016). Without the combined work of Professor Falco, the students who digitized and archived all of this media and subsequently built the page to showcase it, Mr. Cancelli's work may never have been made available to the vast audience of the JHDB. Cancelli is just one of countless musicians whose work and livelihoods have been painstakingly documented and made universally accessible.

Reach

The reach of the Jazz History Database now extends far beyond New England. Google Analytics results for jazzhistorydatabase.com show that, in March 2020 alone, the website was visited by users in 58 different countries across 6 continents. Since 2010, jazzhistorydatabase.com has been visited by nearly a hundred thousand unique users from 178 countries. While these statistics stand for themselves, it is worth noting that the JHDB serves a very niche audience, and to date has not had the resources to localize the majority of its content in any language other than American English.

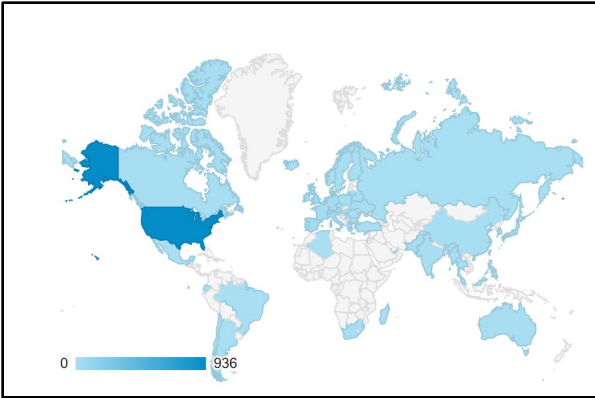


Figure 5.1. New users by country world map (Mar 1, 2020 - Mar 31, 2020). Source: Google Analytics

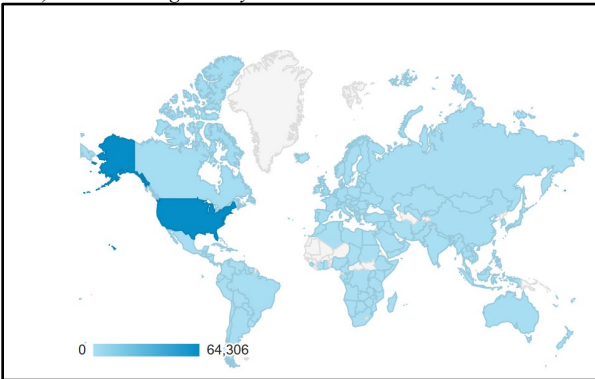


Figure 5.3. New users by country world map (Jan 1, 2010 - May 1, 2020). Source: Google Analytics

Country ?	New Users ? ↓
	93,503 % of Total: 100.05% (93,454)
1. United States	64,306 (68.77%)
2. United Kingdom	2,941 (3.15%)
3. France	2,139 (2.29%)
4. Canada	1,874 (2.00%)
5. Italy	1,691 (1.81%)
6. Germany	1,679 (1.80%)
7. Japan	1,335 (1.43%)
8. Brazil	1,262 (1.35%)
9. Spain	1,222 (1.31%)
10. South Korea	916 (0.98%)

Figure 5.2. New users by country, top 10 countries (Jan 1, 2010 - May 1, 2020). Source: Google Analytics

Day to Day Operations

Every contributed artifact or piece of jazz literature is subject to a predefined workflow prior to publishing. Any original media, especially artifacts stored in obsolete mediums (for instance, a cassette tape), must first go through a lossless digitization process. The digitized media is then stored in multiple redundant digital storage solutions and uploaded to Dropbox (a cloud storage solution), to ensure preservation of the media in any kind of outage or disaster. During this process, the JHDB seeks permission from the respective copyright owner(s) to serve the newly preserved media or literature on the website. Assuming permission is granted, an edited copy of the media is then placed on a webpage in the JHDB website for display. Any copyrighted content for which permission is not explicitly provided is edited out prior to publishing, but the original digitized work is still preserved, and in most cases shared with the owner.

This workflow is the collective effort of students, work-study employees, and Professors Rich Falco and Mike Drnek. Professor Falco oversees the entire process, managing the backlog of contributions and allocating students and employees to tasks periodically. He also acts as the point of contact for external contributors and other third parties. Professor Drnek oversees the technical implementation of the archive and the website.



Figure 6. Standard Workflow for Jazz History Database Collections.

Every part of the process is handled by a different part of the JHDB team. The digitization process is fulfilled entirely by students. While some of this work is done by Federal Work Study funded students, Professor Falco also hosts an inquiry seminar (“Lorraine Feather,” n.d.) where students digitize content for JHDB in a classroom setting. Students not only learn about preservation and familiarize themselves with the JHDB - they learn how to digitize various analog media solutions, edit images and audio, and create, edit, and transcode video media using industry standard tools. During digitization, Professor Falco may contact the appropriate copyright owners to request permission to host their media on the site.

The webpage building process is handled entirely by work-study employees. These employees incorporate new contributions into the website once they have been digitized. The process involves double-checking the media to make sure it is “web-ready,” and building a web page from scratch to showcase these materials based on the predefined design standards of the website.

With both Professor Falco and Professor Drnek retiring, and major funding cuts due to COVID-19, it became critical to the survival of the JHDB to reduce the time, resources, and technical skill required to publish the JHDB’s extensive backlog of digitized content. Our work has allowed non-technical volunteers of the JHDB, including Professor Falco himself, to quickly build and publish tens of pages that would otherwise have taken the span of weeks or months to get published with limited staffing and funding.

The Problem

Unlike the digitizing and editing of the media, which is an inherently manual and individualized process, building web pages to showcase the content does not need to be. With two sessions per year of Professor Falco’s inquiry seminar digitizing media at an impressive rate, the biggest bottleneck in publishing this content is the webpage building process. Furthermore, even though many pages on the site share a similar structure, individual rebuilding of these pages leads to inconsistencies in the format, aesthetic, and design of the online museum. Take for example the biographies of Lorraine Feather and Amanda Carr (“Lorraine Feather,” n.d.; “Amanda Carr,” n.d.). Both of these biographies feature the same general content - a bio photo, a text biography, and sections containing galleries of images, videos, and audio recordings. Some of the differences in layout may be intentional - such as the ordering of the content, or the relative sizes of the videos and images. Other elements however, such as text sizes, header and link formatting, and the wrapping of text around the bio photo differ in ways that do not reflect design decisions. On smaller screen sizes, the paragraph spacing on Amanda Carr’s biography breaks, and the header for Lorraine Feather’s biography is missing altogether.

Thus, we identified two core problems with the page-building process: large amounts of unnecessarily repeated manual work for small pages, and inconsistencies introduced by human error across the output of this repeated work. It is reasonable to expect that students with different levels of technical and design expertise would produce different results while trying to create similar content. By

creating a system that takes in the media and feeds it into a reusable template, the repeated aspects of the process are removed and outputted pages with the same structure are uniform, and not subject to formatting errors.

Through the development of the prototype in the D-term 2019 independent study, we discovered an additional problem that we were already partially addressing. Due to the amount of time involved in the page-building process, there was an extensive backlog of pre-digitized media that needed to be built into pages. Much of this media was submitted in a digital format by contributors, which posed the question: if some contributors were already able to upload this media to the JHDB Dropbox, and students are simply inputting the media into a web form to generate the pages, why not give contributors access to create the pages themselves? As such, our third problem became how to enable contributors with pre-digitized content to create pages themselves.

Approach: Why Start From Scratch?

The solution that we ultimately accepted was building a custom web application that could interface directly with the JHDB's existing infrastructure. This primarily involved using the JHDB's Dropbox Business account for media storage, and outputting static web pages that would integrate with the existing website's theme, headers, and footers. These specific needs are what originally informed the decision to create custom software from scratch, as opposed to integrating with an existing content management system such as Wordpress. While Wordpress provides similar core functionality for uploading media and creating/hosting pages, it adds the overhead of a database for storing and accessing that content. It also did not provide the flexibility we needed to integrate with other content storage formats, or provide a simplified user experience for the media uploading and page creation processes ("WordPress," n.d.). Familiarity with the involved technologies further solidified this decision. Our team was much more comfortable designing and creating a javascript-based web application than integrating with the PHP-based ecosystem of Wordpress, which would have entailed learning an entirely new programming language (and one that is notoriously inconsistent syntactically) (Hafiz, 2010). While we did explore other less popular content management systems, all posed similar concerns regarding the learning curve and integration with the JHDB's established tools/workflows.

Previous Work

This project was first attempted in 2016 by a Computer Science (CS) student working part-time for the JHDB. While the project attained minimum functionality, the user experience was not tailored to the target audience and the technical implementation had some fundamental problems. While the project was not documented, and the specifics are not relevant to the scope of this paper, we were able to infer some pitfalls to avoid between an archive of old test data and some anecdotal information from Professor Falco. The system relied on a filesystem-based data storage solution - files stored in specific directories were indexed by PHP web pages to build the user interface, which made several assumptions on the user's input. The resulting tool was slow, and too confusing to use for a wider audience. Early on, we decided to use a database for storing user/contribution data as that would integrate more naturally with a web-based application. We also focused the design of the interface very heavily on our target user base - it was critical that the tool not only be usable and well-documented, but also intuitive for a non-technical audience.



Figure 7. File storage structure from 2016 attempt at a JHDB content management system.

As previously mentioned, the preliminary proof-of-concept for this project was completed outside the scope of this IQP in the form of an independent study project (ISP) in D term 2019. The scope of the ISP involved preliminary identification of the problem and technical constraints, as well as the development of a rough prototype to ensure the technologies we planned to use could accomplish the baseline functionality we needed. Specifically, we wanted to create a custom web form where users could upload media files, select specific image/audio/video files and enter biographical information. This web form would subsequently generate a web page using that media/text and display it to the user.

While we were able to accomplish most of these goals, uploading files to the JHDB Dropbox proved unexpectedly complicated from a technical standpoint. Due to budgeting restrictions, we could not provide every contributor with a Dropbox Business account, and addressing the nuances of uploading very large files over potentially unstable internet connections pushed a fully custom solution out of scope. Between the completion of the ISP and the start of the IQP, we discovered a way via the Dropbox API to programmatically generate a File Upload Request link. This feature of Dropbox had been used manually in the past by the JHDB to collect files from contributors securely, and allowed us to outsource the technical aspects of handling user file uploads to Dropbox. Automating it proved surprisingly trivial, and thus we were able to overcome our last major technical hurdle for the minimum viable product.

Methodology

Objectives

Our goal was to empower developers and remote correspondents of the JHDB with the ability to contribute to the JHDB website remotely, seamlessly, and with little technical skill requirements. The reputation of the website and the preservation of various jazz artifacts from multiple countries were at

stake. To accomplish our goal, we built a web-based tool (“contributor portal”) where developers and remote correspondents may upload media into a simple web form, which outputs web pages from reusable templates and publishes them at <https://global.jazzhistorydatabase.com>. We produced the following list of requirements in collaboration with Professor Falco (our primary stakeholder) to evaluate our achievement of this goal, in no particular order:

- The contributor portal [it] must be available online at all times.
- It must require little familiarity with computers and minimal technical expertise.
- It must feature various sources of aid and clarification to the user.
- It must allow the user to upload text, image, audio, and video media into pre-build web page templates.
- It must allow the user to request approval to publish a collection of web pages. Once approval is electronically granted by an administrator (such as Professor Falco), the collection shall be published at <https://global.jazzhistorydatabase.com>.
- It must allow the user to create and maintain multiple collections.
- It must sandbox users to their own particular ‘branch’ of the global JHDB, such that a correspondent will not have access to another correspondent’s collections, unless they express that they work on the same collections.
- Its source code and repositories must include extensive documentation, to facilitate onboarding future developers into the project.
- Lastly, the portal must protect the integrity and identity of the JHDB and its members, such that no personally identifiable information is unintentionally publicized, and no remote correspondent could accidentally damage the portal or the website.

Our overall objective can be summarized as the design, development and evaluation of a feature-complete contributor portal (i.e. allowing users to go through the full process from login to publishing a page).

Target Audience Evaluation

From the start, a core focus of this project was designing an intuitive interface around our target user base. As discussed in the Previous Work section, prior attempts at this project failed, despite meeting basic technical requirements, due to a lack of usability for the target user base. To identify the needs of our audience, we employed a purely qualitative approach. Since our audience was primarily non-technical, we concluded it would be easier for them to communicate their concerns in a more anecdotal/conversational format, as opposed to answering predefined questions on their expectations of the functionality.

We worked primarily with Professor Falco for this portion, with occasional input from Professor Drnek and Tom Bellino. As our primary stakeholder, Professor Falco showed videos and demos of our earlier prototypes to prospective contributors to get their feedback, and communicated the takeaways of those conversations back to us. This allowed us to focus on balancing the development of the technical pieces of the tool with addressing the user experience bugs, without having to maintain communications with all our stakeholders directly. From these conversations, we identified a few key focus points:

- **Phrasing** - our target audience consisted largely of jazz musicians and historians without technical backgrounds. As such, clear and unambiguous phrasing of instructions and button/form labels were critical. Simple details such as referring to built collections as “contributions” could be misconstrued to imply monetary contributions.
- **Layout** - we initially planned to base our design on Google’s Material Design specification (“Material Design,” n.d.). While most of the design principles such as visual consistency and responsiveness carried well, we opted to stray from the guidelines in certain key areas. Most notably, icons that have a clear and consistent meaning for frequent users of modern websites/apps, such as the hamburger menu icon, did not effectively communicate the associated functionality for non-technical users. This was addressed by clearly labeling buttons with their functionality to avoid any possible confusion/ambiguity.

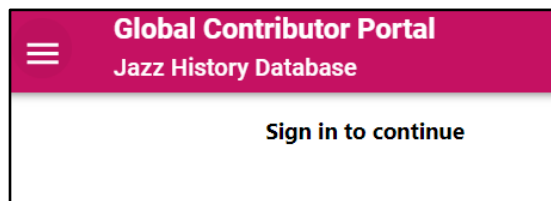


Figure 8.1. Old prototype for login flow. This is standard in many modern web applications, where users are used to clicking on the hamburger button and logging in via the app drawer, but is ambiguous to an unfamiliar user

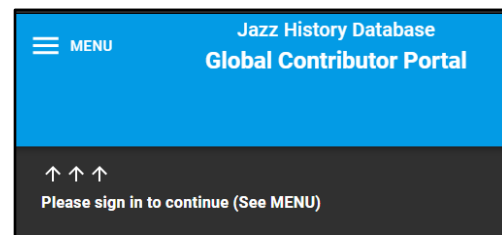


Figure 8.2. Updated login flow. Clear visual cues and labels communicate what the user needs to do, while maintaining the standard layout for users with predefined expectations.

- **Colors** - by default, the UI component framework we used consists of a light theme with bright, warm colors to denote functional UI elements (navigation bars, buttons). While there is nothing inherently wrong with these color choices from a technical standpoint, we found that they did not give off “jazz vibes.” We opted to change this in favor of the dark theme to reduce eye strain along with light blue highlights to match the color palette of jazzhistorydatabase.com (“Dark Theme,” n.d.). We received immediate positive feedback on this change, as the cooler, less harsh color scheme fit with predefined associations that some users had with the colors (light blue equals jazz) and provided an overall “clean” aesthetic.

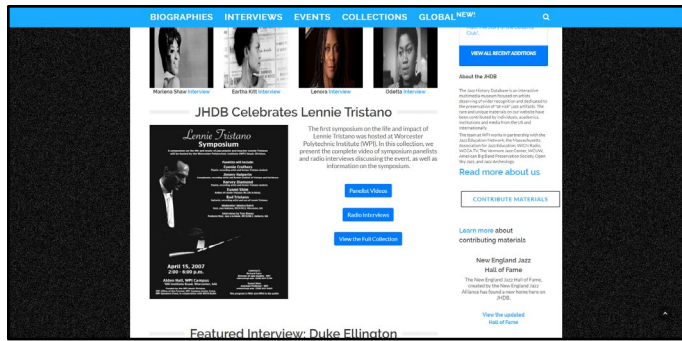


Figure 9. Light blue color palette with dark grey background at jazzhistorydatabase.com.

UI/UX Design

As we have established, the user interface and user experience as they pertain to our target audience were major priorities in the design of the final product. As discussed in our target audience evaluation, several color and layout choices were directly influenced by feedback from our target audience communicated through our key stakeholders. For visual components, we utilized the Material-UI framework (“Material-UI,” n.d.). This provided us with pre-made UI elements with default layouts and aesthetics in accordance with Google’s material design specification.

Arguably the most important aspect of the user experience is the landing page. If the web application does not seem simple, friendly, and inviting, a non-technical user may feel intimidated or simply uncompelled to even begin the process of exploring it. In *The Design of Everyday Things*, Don Norman discusses in depth how the affordances of computer software are often unclear and downright confusing to the average person (Norman, 2013). Thus, we wanted to limit the affordances on the landing page to a bare minimum, and provide clear signifiers for the user to quickly identify the first steps they should take: watching the getting started video, and logging into the contributor portal.

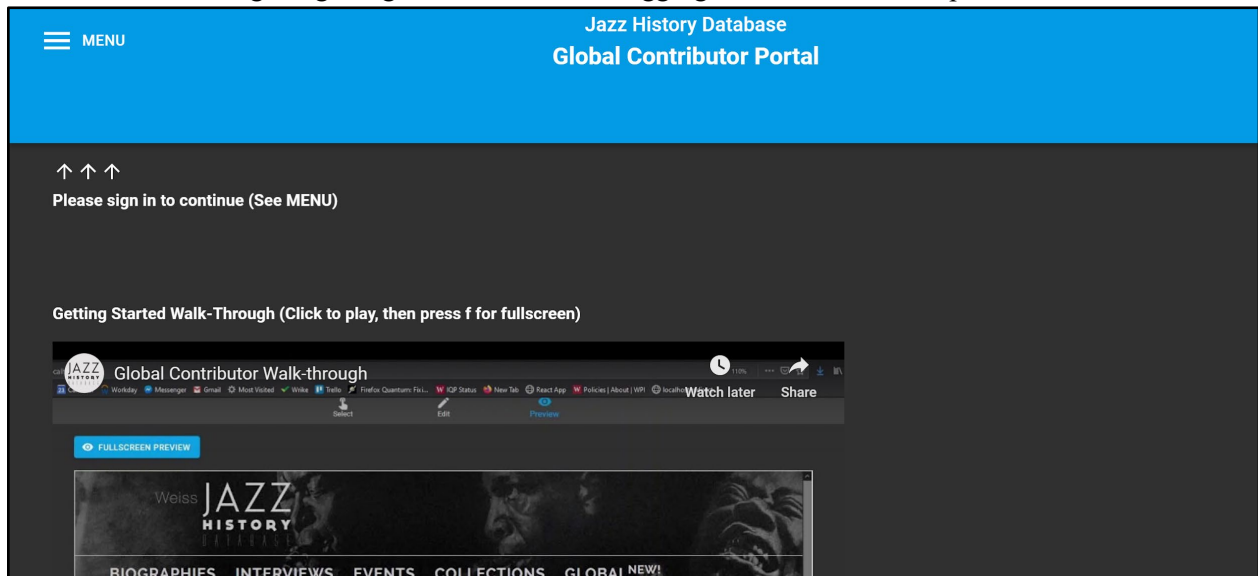


Figure 10. Global Contributor Portal landing page.

The landing page accomplishes just that. The title bar immediately tells the user where they are (The Global Contributor Portal), and clear visual cues introduce the user to the menu drawer if they are

not familiar with that design pattern. The video walk-through is embedded right on the landing page, so a user that is unsure of how to log in can receive the instructions they need in a visual and conversational format if they are unsure of how to proceed.

The menu drawer is also minimal. Notably, it has a labeled button to close the menu. While more experienced users may know that clicking outside the drawer will dismiss it, and this expected functionality persists, that affordance is clearly communicated to users that may not be familiar with those typical behaviors. The only other visual elements are the “Sign In” button, and a help button that will prompt the user to watch the instructional video and reach out via email for assistance if needed. Clicking the sign-in button will show a pop-up to the user explaining what will happen next before they are redirected to the login screen, and giving them the option to cancel that action if needed (e.g. if they realize they have not retrieved the credentials for their account and want to reference the video).

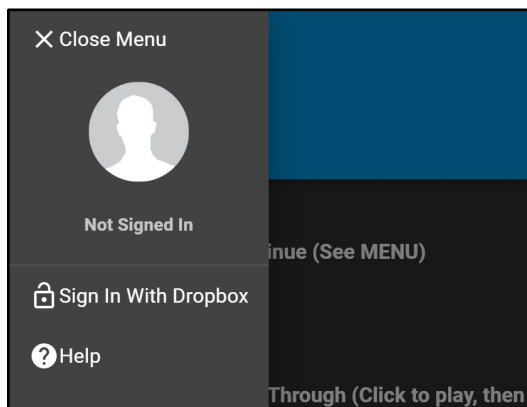


Figure 11.1. Portal menu drawer.

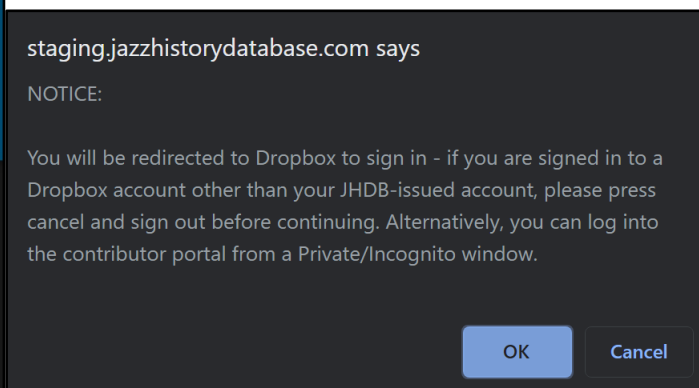


Figure 11.2. Portal notice that the user is being redirected for login.

These same principles persist after signing in. The application loads with the information tab active. Notably, all of the UI elements the user has been introduced to remain in view. The user is greeted with some friendly welcome text, and prompted to return to the video (which will now start at the point immediately after login in the tutorial, where they may have previously paused it). A brief text description provides an overview of the portal’s features, explaining the new tab bar that has appeared under the title bar for users that are compelled to explore on their own. The user is again invited to reach out via email for assistance if needed.

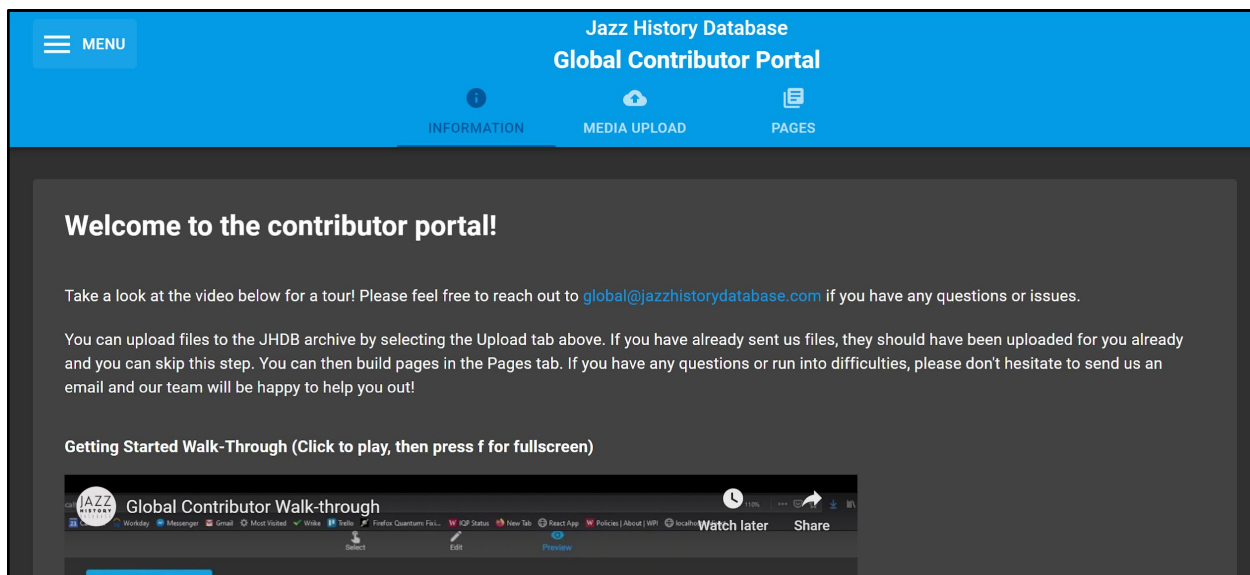


Figure 12. Contributor Portal Information Screen immediately after logging in.

These key design principles are upheld throughout the application, and the details of the application’s functionality and end-user workflow are discussed further in the results section.

Development Workflow

To ensure overall completion of all our objectives alike, our team adopted a standardized development workflow from the beginning of the project. As a very small team, strictly adhering to an industry standard project management framework would have created unnecessary overhead. Thus, we developed a more informal workflow borrowing some principles from Agile and Scrum as defined by Jeff Sutherland in his book, *Scrum: The Art of Doing Twice the Work in Half the Time*, in combination with industry standard tools for collaborative software development (Sutherland, 2016). We utilized GitHub to host our code and working technical documentation, and made use of the built-in Issues and Projects features to track outstanding tasks and current progress. While we scrapped the point-based elements of Scrum, we held daily check-in meetings and frequently prioritized our backlog of features. Tasks were core parts of our workflow to ensure we kept ourselves on track and were always working on the highest value features and bug fixes (with value defined as progress towards attaining our objectives for the project). This maximized our progress on the overall end-user experience and reaching our feature-complete milestone.

With D-term moved online, our communications platforms also became a key element in our daily workflow. As a team of two, we communicated efficiently via Facebook Messenger for convenience. We hosted Zoom calls for most daily check-ins and other meetings where verbal communication was more convenient or practical than text. We utilized the JHDB’s existing Slack workspace to communicate with Professor Drnek and the JHDB staff. For communication with Professor Zizza (our advisor) and Professor Falco (our key stakeholder), we mainly employed emails for written updates and communications, and for scheduling regular face-to-face meetings via Zoom (“Naming,” 2020). For added convenience, we held a recurring Zoom meeting as a dedicated virtual “room” for our meetings, accessible via a memorable short link with bit.ly.

Technologies Used and Supporting Systems

We utilized a number of established technologies and frameworks in developing the portal to maintain our focus on the core design and functionality of the application and work efficiently. In addition to the development of the tool itself, there were also several considerations regarding the environment on which the tool/website would run and associated technical hurdles. The detailed technical specifications of the production environment and the process of setting it up are outside the scope of this paper, as they are tangential to our objectives. These systems were critical to the success of the project however, and thus we provide a high-level summary below:

Production Environment/Infrastructure

- **Google Cloud Platform**
 - **Dev, Staging, Production** (“Naming,” 2020)
 - Isolating development, testing, and production (user-facing) environments is a concept very familiar to site-reliability engineers. We adopted the standard model with three environments: dev (short for “development,” a less secure environment with no sensitive data to facilitate working on and testing the code for the portal), staging (a mirror of the production environment where we could deploy and test changes to find any bugs introduced by the build/deployment process before rolling out to users), and production (the user-facing environment, where all of the actual data is stored).
 - **App Engine (Standard Environment)** - *App hosting platform as a service (PaaS). Hosts multiple instances of the contributor portal server, which in turn serves the user-facing web client. The generous free tier allows us to simultaneously host multiple versions of the portal without incurring cost (“The App Engine,” 2020). This platform also scales well should there be a need to increase the capacity of the software in the future. Combined with the rest of the Google Cloud suite, it also provides useful features such as logging the application’s output, removing the need for us to develop a custom solution for tracking usage and errors and allowing us to quickly diagnose and troubleshoot issues in our production environment.*
 - **Cloud Build** - *Detects changes to the master and staging branches in our GitHub repository and automatically builds and deploys the latest version of the application. Build configurations are specified as YAML files that list the steps involved in building the application. Our build process involves building and packaging the client and server code, copying the corresponding credentials file, and deploying to App Engine (“The App Engine,” 2020). While this is technically only five commands, executing these steps automatically and in the cloud saved us from many hours of dealing with the nuances of manual deployment.*
 - **Cloud Firestore** - *A real-time database-as-a-service accessible via a REST API (“Cloud Firestore,” 2020.). Firebase provided us with an authentication and permissioning system out of the box, so that we could focus on developing the application from the early prototype stages instead of getting caught up with the nuances of a self-hosted SQL server and developing our own authentication and permission systems. Firestore (the database service from Firebase we employed) handles permissions via database rules*

that are expressed in plain Javascript (Matticoli & Varella, 2020). This allowed us to properly secure access to our user data and handle authentication via a few lines in our client-side code. Notably, Firestore employs a serverless model for accessing application data, so the server and client code can interact with the database individually (using the Firebase Admin SDK and the Firebase Web SDK respectively) and eliminate the need for defining API routes for each piece of data/action.

- **Dropbox** - Since the JHDB has a Dropbox for Business plan, we are utilizing Dropbox as the primary means for file storage. This allows us to store unlimited files and data without incurring any additional cost. For the short term, we are also utilizing Dropbox as the CDN (Content Distribution Network) for the media on the published pages. While this is not a scalable solution, it allowed us to provide the full user experience of accessing a published page without having to worry about processing and hosting media files within the scope of this project.
- **Self-hosted Virtual Private Server (WPI Research Cluster)**
 - **The Server** - WPI's Academic Research Center (ARC) provided us with a virtual Ubuntu 18 server with 8GB of RAM, a quad-core vCPU, 100GB local storage, and 2TB of persistent hourly snapshot storage via a network file system. This allowed us to create a customized, fault-tolerant production environment without having to pay the substantial monthly costs of a comparable VPS in Google Cloud's Compute Engine.
 - **Docker+Portainer** - The Docker container system alongside the Portainer user interface for managing said containers allowed us to deploy supporting applications (such as our web server and monitoring solutions) in a scalable and reproducible fashion ("Docker," n.d.; "Portainer," n.d.). Each application is defined in a YAML file that details the "image" (a pre-built collection of binaries and dependencies for an application), environment and configuration data. These configurations can be deployed to any Portainer instance with a pasted link to the GitHub repo and a single click. This meant our entire server environment could be represented through text files in GitHub repositories, allowing new maintainers to quickly understand and maintain our system.
 - **Nginx** - Nginx is a web server that is able to handle many concurrent connections. For our purposes, the nginx server hosted all of our domains, including a mirror of the original static content at <http://www.jazzhistorydatabase.com/> served over HTTPS so our templates and published content could function on secure systems such as Google's App Engine. The Nginx server also included configuration for each subdomain of jazzhistorydatabase.com, and served as a reverse proxy and static host for the global pages (those published from our contributor portal) to create a seamless experience across all the different components of the website.
 - (All items listed below represent subdomains of jazzhistorydatabase.com in the form <item>.jazzhistorydatabase.com)
 - portainer - Reverse proxy to Portainer web portal, served over HTTPS
 - contribute - Reverse proxy to jhdb-cms.appspot.com, the production contributor portal
 - staging - reverse proxy to staging-dot-jhdb-cms.appspot.com, the staging environment for the contributor portal

- *dev* - reverse proxy to *dev-dot-jhdb-cms.appspot.com*, a development deployment environment for the contributor portal to test build process changes
- *global* - static hosting of published pages from the contributor portal synced from Dropbox
- **Amplify** - A service of Nginx that provides monitoring, analytics, and alarms for our Nginx server. The Amplify client runs in the cloud, and provides a web interface at *amplify.nginx.org* that displays information sent up from an “agent” application running on our server. This allows us to view website traffic and error data in real-time, and in the event of a server or network outage, will detect that the agent is not communicating and notify us immediately so that we can address the problem and minimize downtime.

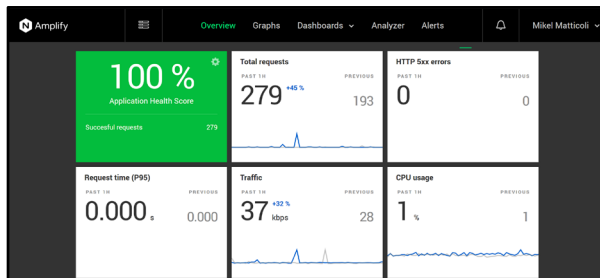


Figure 13.1 Amplify dashboard shows at-a-glance system health.

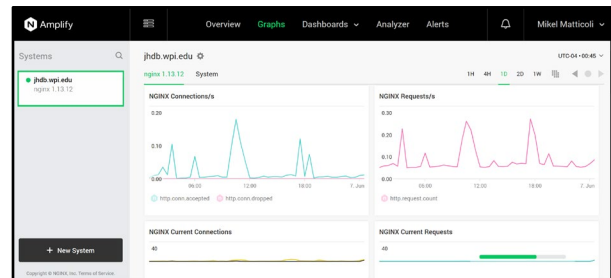


Figure 13.2. Amplify real time usage statistics and analysis.

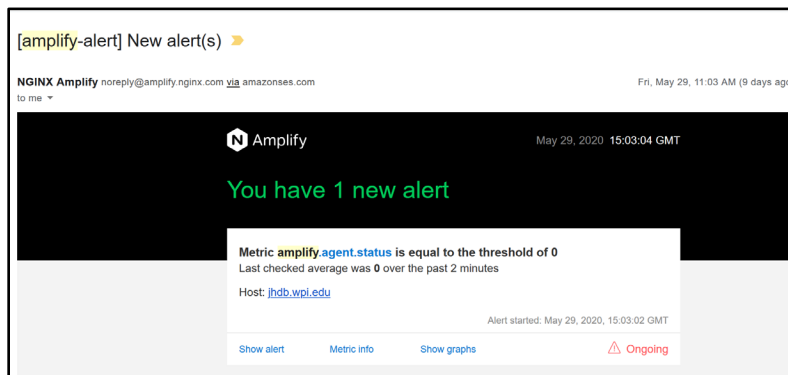


Figure 13.3. System outages trigger immediate email alerts from Amplify.

- **Dropbox Sync** - Since the contributor portal interfaces with Dropbox to store media, we opted to use the same system to publish pages. The PHP pages generated by the contributor portal are saved to Dropbox, and an instance of the dropbox service running on our server syncs these files down in real time allowing for effectively instantaneous deployment of pages.

Technical Implementation

Component Hierarchy (User Interface Breakdown)

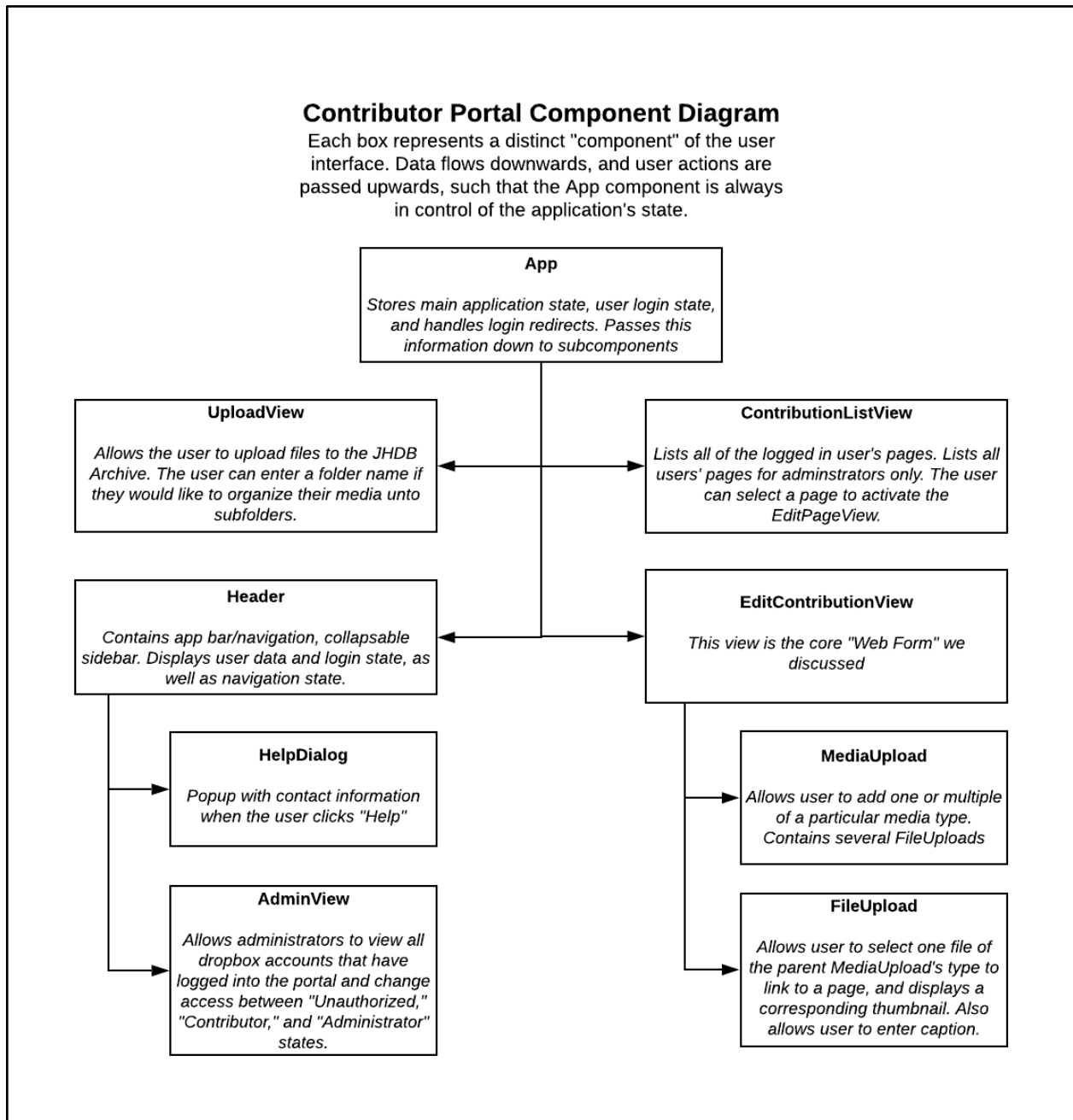


Figure 14. Component hierarchy of the portal's client-side React.js code.

We organized the client-side code of the contributor portal into a hierarchy of React.js Components. At the top of said hierarchy lies the App component, which serves as a container for the rest of the client user interface. All themes and most styles used throughout the portal are defined in the App component, then passed down to active components. It handles the portal's authentication routines with Dropbox and Firebase (once the user logs in). At any place in the portal, the App displays two main child

components: the site Header and the user's current page on the portal. We will traverse the components in this hierarchy in order from top to bottom.

Header

The Header component houses the portal's title, a group of tabs underneath, and the portal's menu on the top left. Opening the menu shows the user their current profile if they are logged in. Otherwise, the user may navigate to this menu to sign into their JHDB-issued Dropbox account. The menu also allows navigation to the AdminView (short for Administrator View) and the HelpDialog, which provides contact information and special thanks to everyone who contributed to this project. The option to navigate to the AdminView will not be visible until the user signs in with an Administrator account.



Figure 15. Contributor Portal Header Component.

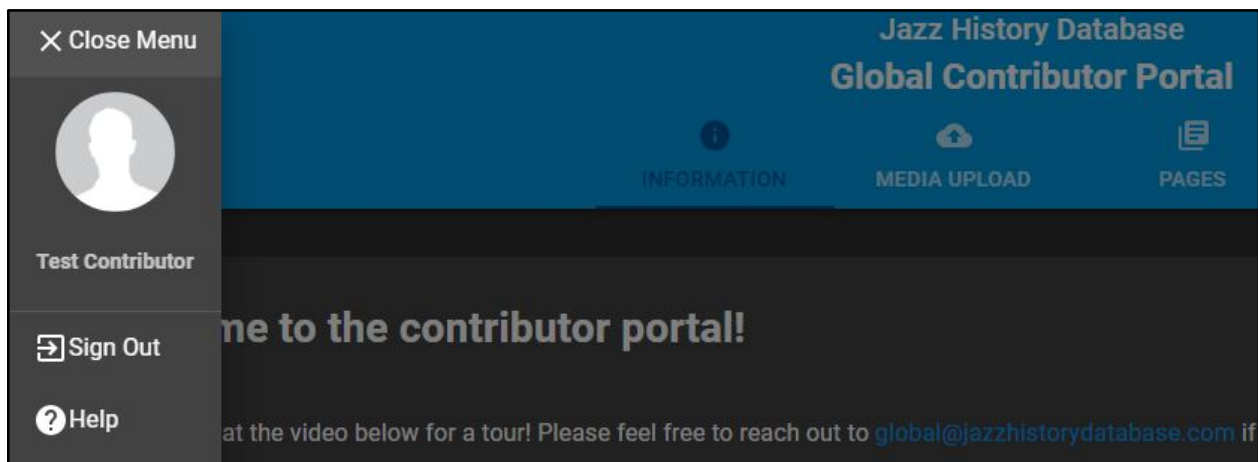


Figure 16. The menu on the top left of the Header, as it appears when opened. The gray app drawer on the left pulls out from the left of the screen and darkens all other components.

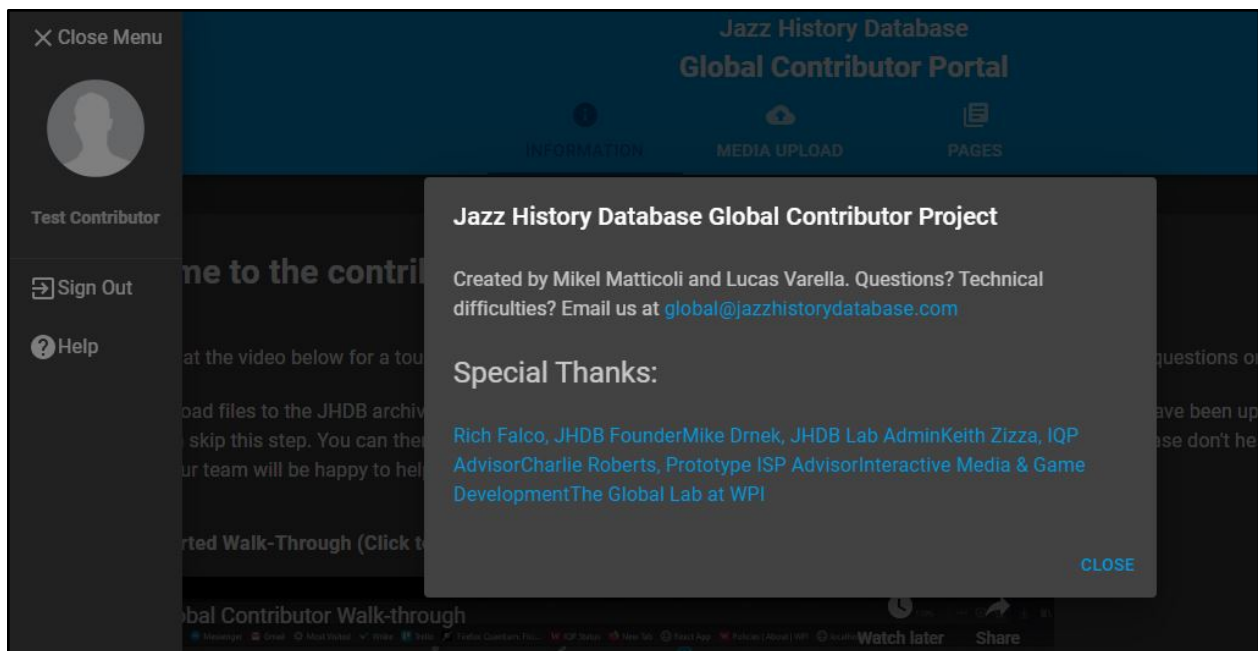


Figure 17. The Help Dialog displayed once the Help button is clicked in the website menu. The user currently signed in is NOT an administrator, and thus has no access to or knowledge of the AdminView.

The AdminView lists all JHDB-issued Dropbox accounts associated with the contributor portal. This component is intended for use only by members and advisors of this IQP, along with any future developers working on the portal. An administrator account has the power to promote other accounts on the portal to administrators, and to publish pages to JHDB's website. It can also revoke rights from other administrators, and even block an account from accessing the portal. An administrator cannot modify their own access rights to the website. Contributor accounts have access to the portal, but they must request approval to publish pages, and access to this component is prohibited. Unauthorized accounts can no longer access the portal.

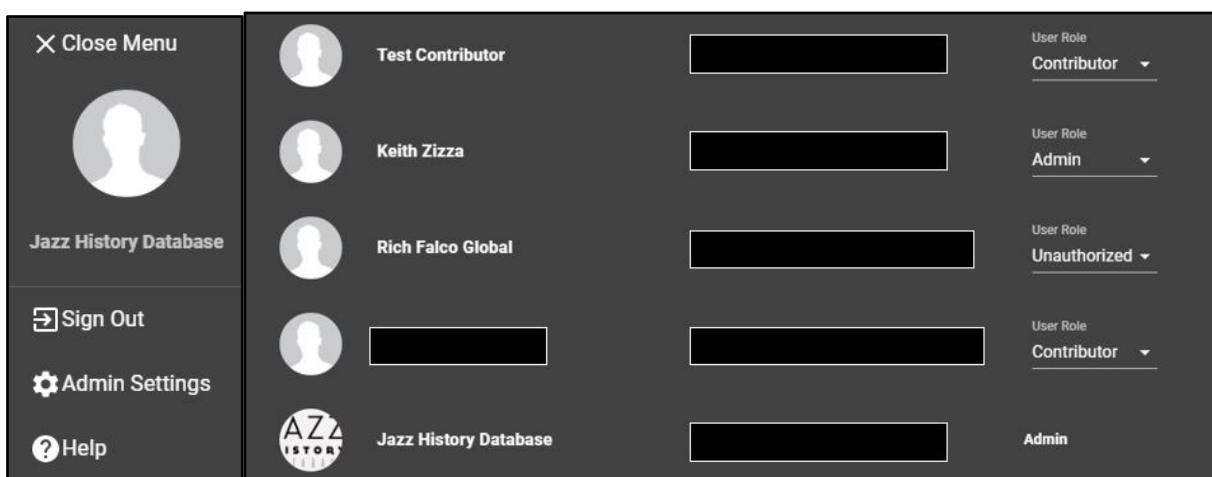


Figure 18. The Menu displayed when the user has logged in with an administrator account on the left. On the right, a sample of the AdminView, with sensitive information redacted. Prof. Rich Falco's account displayed above was likely a duplicate, hence its "Unauthorized" role.

The Header and App components work in tandem to render the current page according to which tab is currently active. The Header contains the tabs that direct the user to other pages, while the App component displays the current page to the user. When the user navigates to another tab, the Header triggers a callback function provided by the App. This callback causes the App to then render the specific component(s) corresponding to the page requested by the user.

The Header features three tabs: the Information tab, the Media Upload tab, and the Pages tab.

Information Tab

The landing page of the portal. Its contents are defined in the App component; It is not rendered by any child components. If the user is not signed in, this page will include a notice to do so. This page features a full video walkthrough of the site, including instructions on how to build a sample collection.

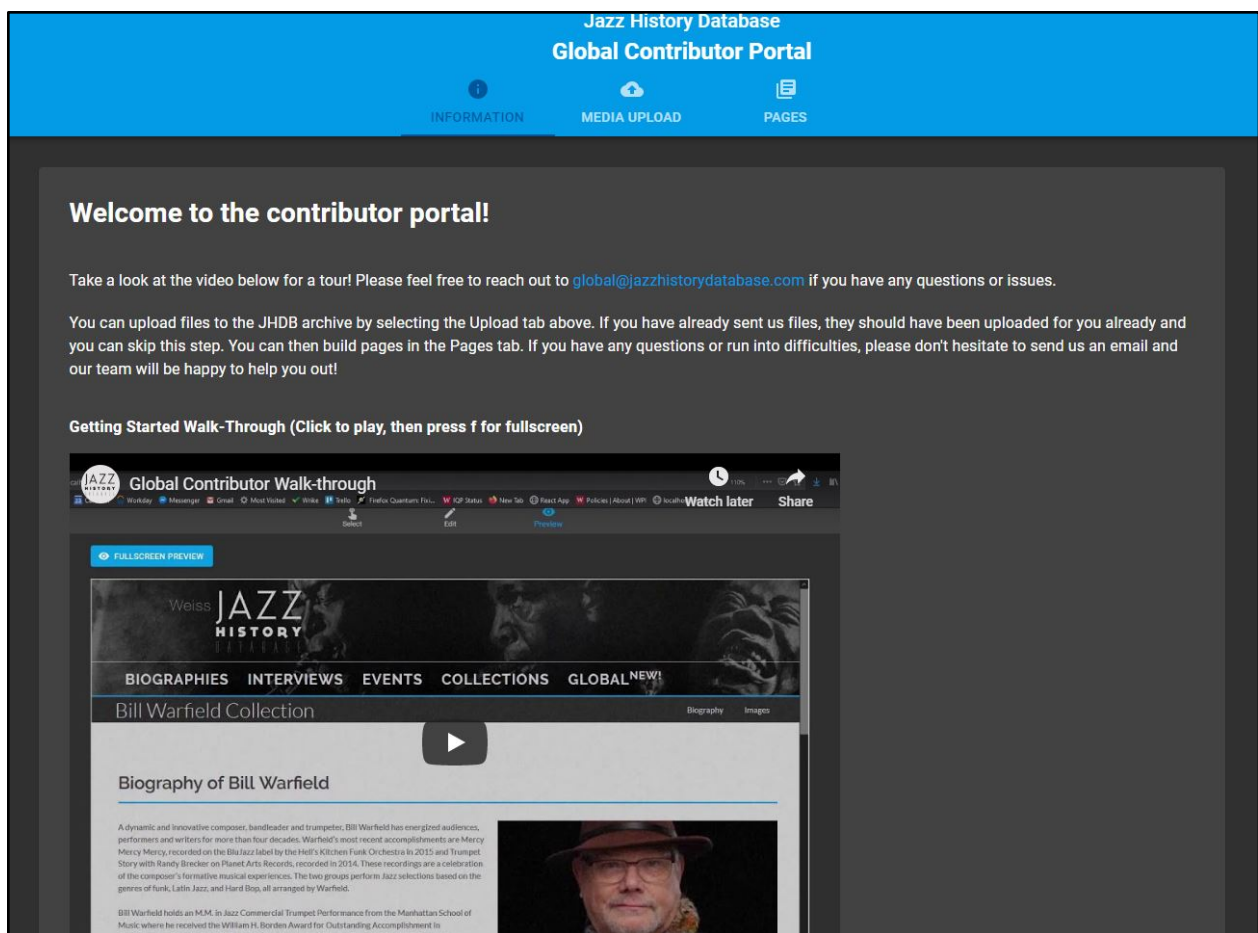
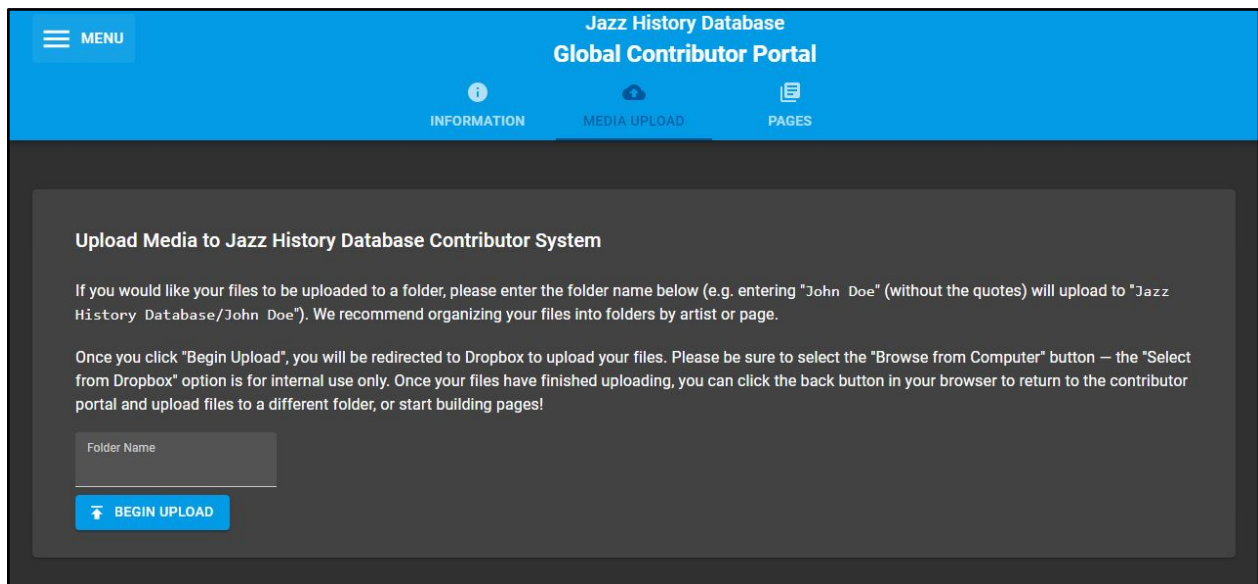


Figure 19. The landing page of the contributor portal, under the Information tab.

Media Upload Tab

Where users may upload their media files to Dropbox prior to building contribution pages. Navigating here renders the UploadView component under the Header. Users may upload to JHDB's Dropbox here, through their JHDB-issued Dropbox accounts. Although any media uploaded here is sandboxed within JHDB's Dropbox, the user may specify folders and subfolders into which the portal should upload their files. Folders and subfolders will be created and reused as necessary. Clicking "Begin Upload" will redirect the user to Dropbox's upload website. Users are instructed to click the "back" button on their browser to return to the portal when they are done uploading media to Dropbox. They may upload as many files as desired before returning.



The screenshot shows the 'Jazz History Database Global Contributor Portal' interface. At the top is a blue header with a 'MENU' button on the left and three navigation tabs: 'INFORMATION' (with an 'i' icon), 'MEDIA UPLOAD' (with a cloud icon), and 'PAGES' (with a document icon). The 'MEDIA UPLOAD' tab is selected. Below the header is a dark gray content area. It features a title 'Upload Media to Jazz History Database Contributor System' followed by two paragraphs of instructions. The first paragraph explains how to specify a folder name (e.g., 'John Doe') for uploads. The second paragraph explains the upload process, including redirection to Dropbox and the use of the 'Browse from Computer' button. Below the text is a text input field labeled 'Folder Name' and a blue button with a white upward arrow and the text 'BEGIN UPLOAD'.

Figure 20. The UploadView of the contributor portal, under the Media Upload tab.

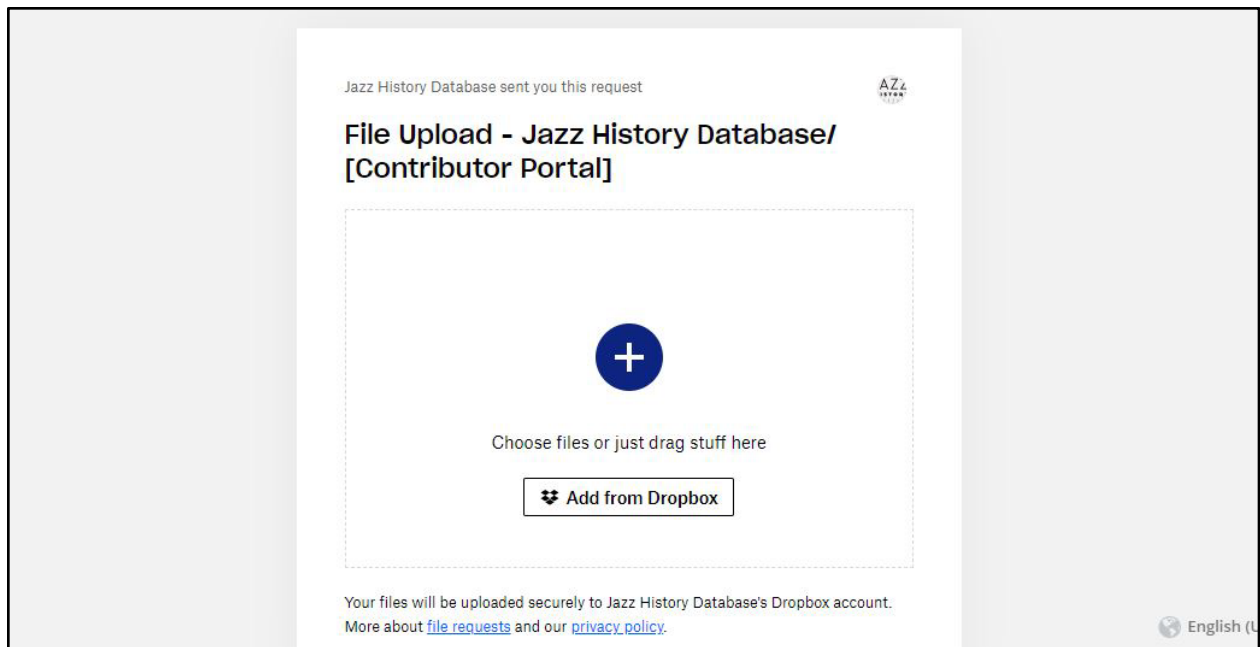


Figure 21. Dropbox upload site, as it looked at the time of this writing. Users are redirected here from the UploadView component.

Pages Tab

Where users may access contributions associated with their account. We expect users will spend most of their time on this tab. Navigating here renders the ContributionsListView component under the Header. This component lists all collections created under the user's account. It allows the user to create a new collection, or select one for work.

Navigating to the Pages tab also triggers a secondary list of tabs: the Select tab, the Edit tab, and the Preview tab. The ContributionsListView corresponds to the Select tab. Other tabs in the secondary header are hidden until the user creates or selects a contribution to work on.

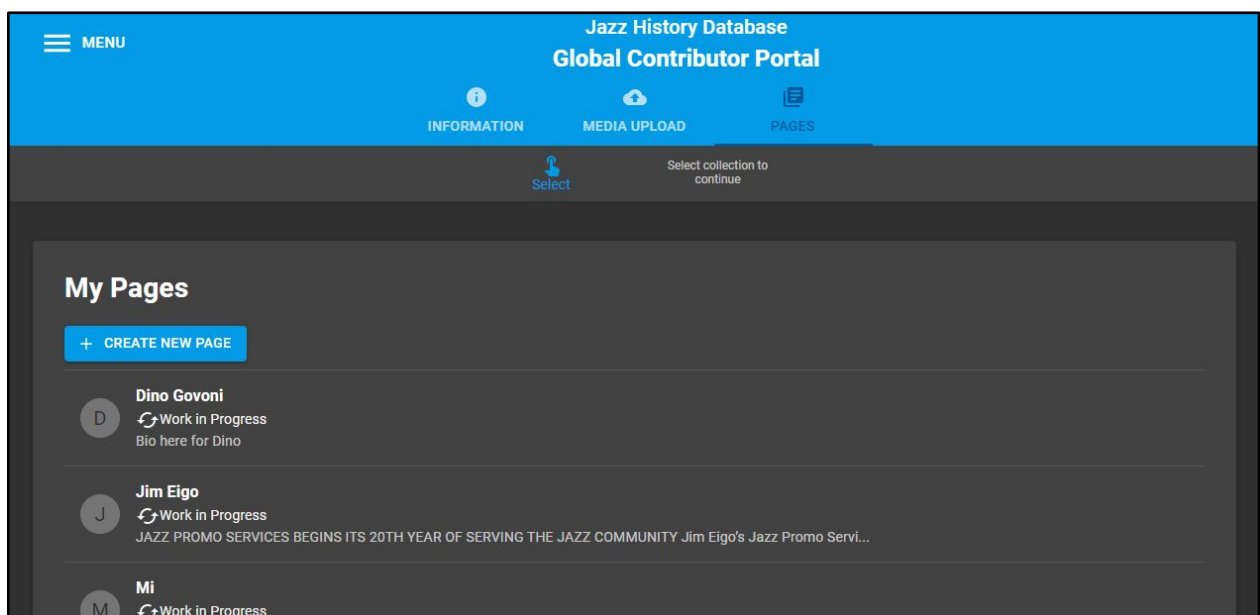


Figure 22. The ContributionListView Component, under the Pages tab, and the Select tab in the secondary tab list. No contribution page has been selected for work.

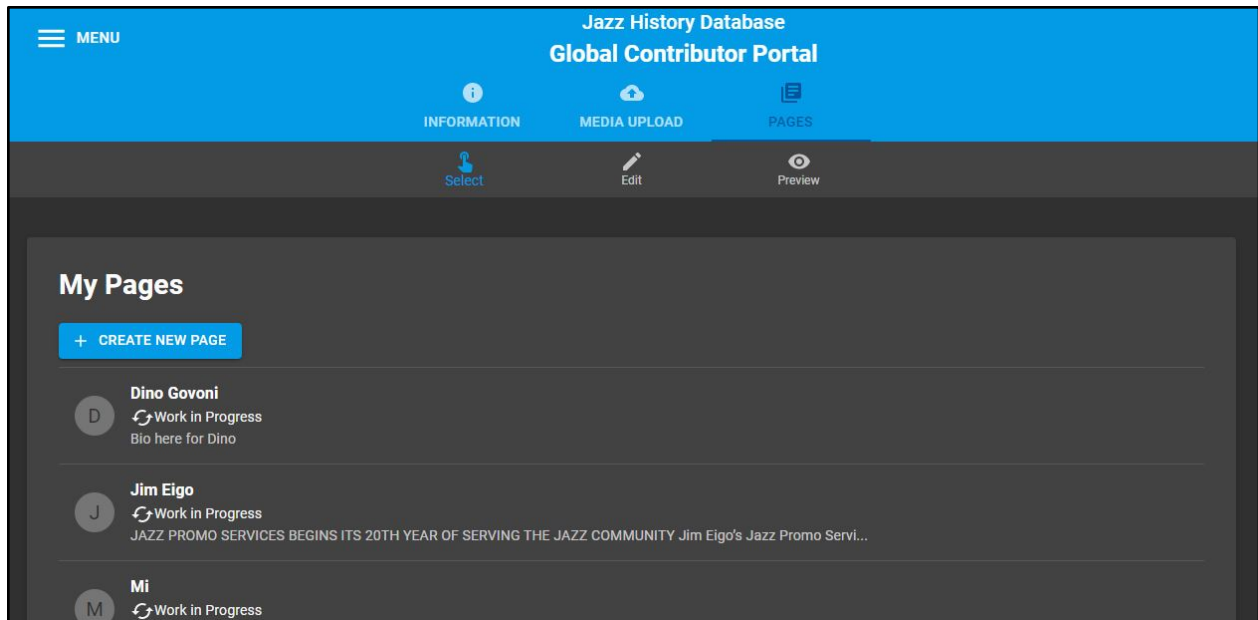


Figure 23. The ContributionListView Component, under the Pages tab, and the Select tab in the secondary tab list. A collection has been selected further down the list, thus all tabs are visible.

Edit Tab

Where users work on their currently selected contribution page. The contents under this tab are rendered by the EditContributionView component, the heart of the portal. As previously discussed, we expect users to visit this tab after uploading all their media, though returning to the Dropbox upload site is only a few clicks away should they need to upload more media.

At the top of the page, users may return to the contributions list by clicking “unselect” (or clicking the Select tab). Once done building this contribution, the user may request approval to publish it. Activating the Request Approval switch marks this contribution as pending approval in Firebase, at which point an administrator may review it. Since only administrators may publish contributions, the Publish switch is disabled for other account types. Note that administrators may deactivate the Publish switch to unpublish contributions, and any user may deactivate the Request Approval switch to rescind their request. Users may delete contributions anytime at the bottom of the Edit tab.

A “Page” is broken down into four major sections: Biography, Images, Audio, and Video. The Biography section is rendered by the EditContributionView. The other sections are instances of the MediaUpload helper component.

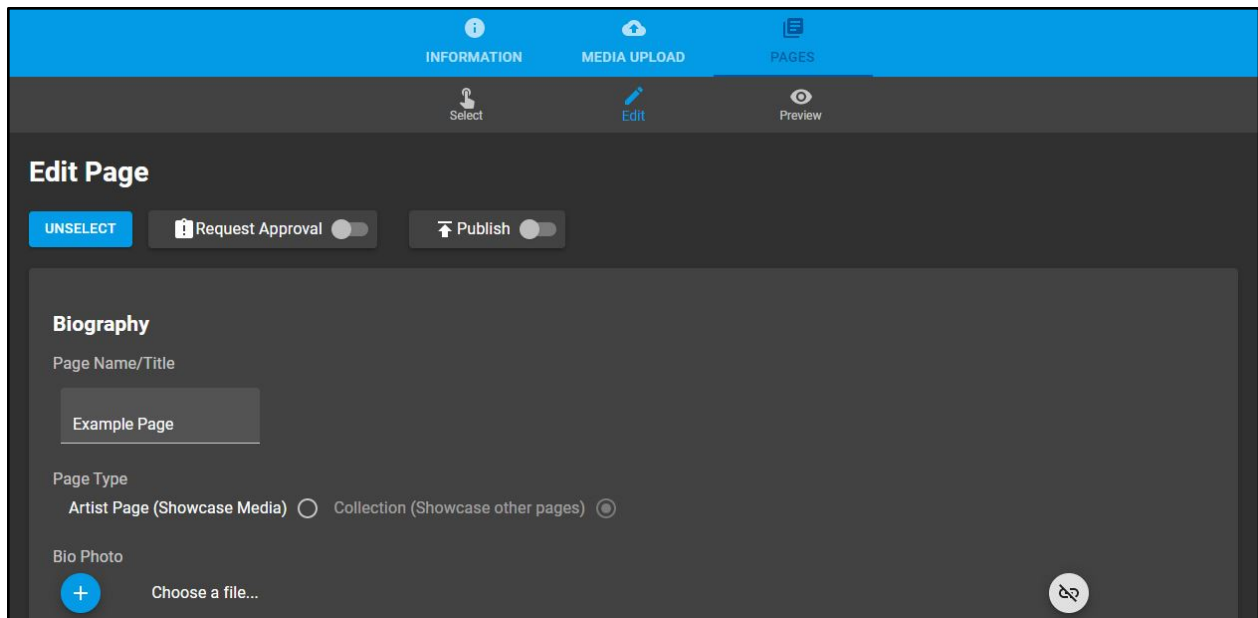


Figure 24. The *EditContributionView* Component, under the *Pages* tab, and the *Edit* tab in the secondary tab list. A new contribution called “Example Page” has been selected.

In the Biography section, users may provide a page title and type, a photo, and the biography text. The page type is purely semantic (though as we will discuss later on, one of our future objectives is to render a different template based on the page type) (“Add Ability,” 2020). The biography photo is not required - the text will occupy all available space in the absence of an image. On the other hand, the text is required, and the Biography section always renders.

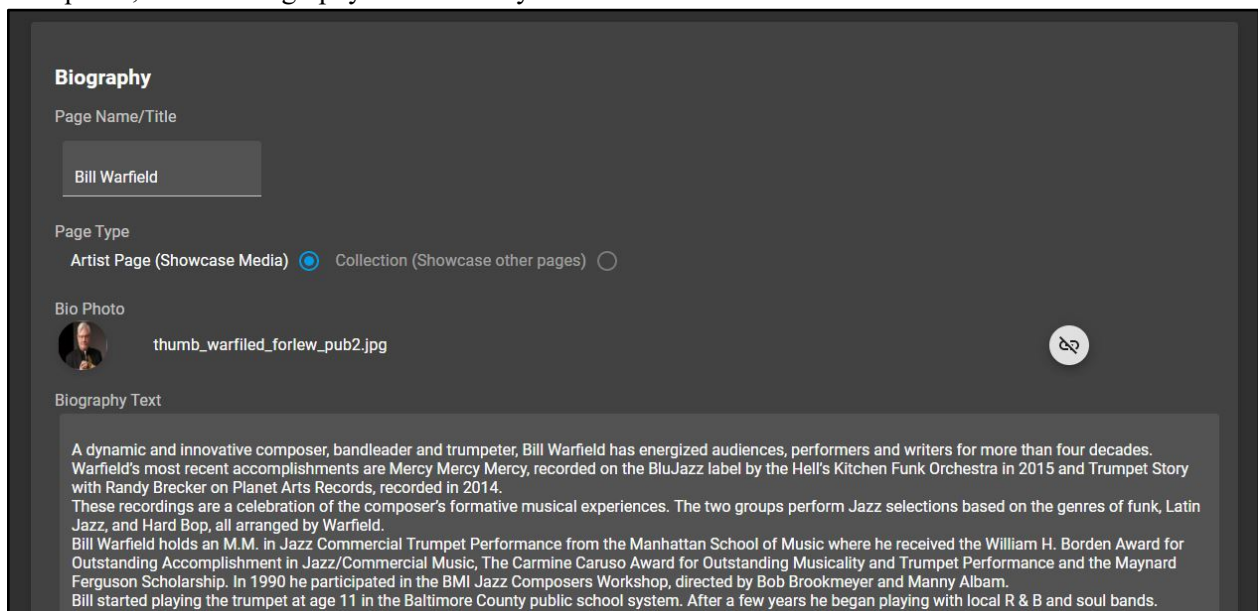


Figure 25. The Biography section of the *EditContributionView*. An existing contribution called “Bill Warfield” has been selected. This is also the contribution used as an example in the video tutorial on the landing page of the portal.

Biography of Bill Warfield

A dynamic and innovative composer, bandleader and trumpeter, Bill Warfield has energized audiences, performers and writers for more than four decades. Warfield's most recent accomplishments are *Mercy Mercy*, recorded on the BluJazz label by the Hell's Kitchen Funk Orchestra in 2015 and *Trumpet Story* with Randy Brecker on Planet Arts Records, recorded in 2014.

These recordings are a celebration of the composer's formative musical experiences. The two groups perform Jazz selections based on the genres of funk, Latin Jazz, and Hard Bop, all arranged by Warfield. Bill Warfield holds an M.M. in Jazz Commercial Trumpet Performance from the Manhattan School of Music where he received the William H. Borden Award for Outstanding Accomplishment in Jazz/Commercial Music, The Carmine Caruso Award for Outstanding Musicality and Trumpet Performance and the Maynard Ferguson Scholarship. In 1990 he participated in the BMI Jazz Composers Workshop, directed by Bob Brookmeyer and Manny Albam.

Bill started playing the trumpet at age 11 in the Baltimore County public school system. After a few years he began playing with local R & B and soul bands. Around that time, his trumpet teacher, Fred Frey, took him to a concert at the Lyric Theater by the Maryland Youth Symphony Orchestra. After hearing the group, Bill auditioned for it and was accepted. The Youth Orchestra rehearsed in the afternoons in downtown Baltimore and Bill spent his Saturday mornings studying "up the street" at the Peabody Conservatory Preparatory School.

As a result of several injuries related to car accidents, at the age of 18, Bill was left without front teeth and the use of a lung. Unable to play the trumpet for a brief time, he picked up a flute and started playing it to restore his wind and breathing.

He entered Peabody but left after one semester because he heard a college Jazz group under the direction of Hank Levy. He enrolled at Towson State College and studied with Hank, spending four years as a featured soloist with the Towson ensemble. The winner of the outstanding Trumpet Player Award at the Quinnipiac College Jazz Festival in 1973, Bill decided that a career as a Jazz Trumpeter was what he wanted. He began frequenting the Jazz clubs in downtown Baltimore and immersed himself in the local jazz scene, performing regularly with musicians such as Sonny Stitt, Mickey Fields, Jimmy Wells and Charles Covington.

Bill left Towson in 1975, but returned as a featured soloist with the Towson Band on their Music Minus One recording $2 + 2 = 5$. His feature solo on Levy's composition "Stillness Runs



Figure 26. The Biography section of Bill Warfield's published page. The portal translates the user's media selections from the previous image into a pre-formatted template.

MediaUpload Component

The Images section transforms a list of images into an animated carousel on the page. Each image supports a small caption. The carousel supports any number of images. It fits up to five images horizontally before relying on the automated animation to display other images. A contribution with less than five images still features the carousel, enlarging images to fit the width of the page. This section is optional - the template omits it when there are no images.

This section is an instance of the MediaUpload component, which represents a section of the template dedicated to a specific type of media. In this case, the MediaUpload component represents all images selected for this contribution (with the exception of the bio photo, unless the user explicitly adds it to the list). It features a “++ Bulk Add” button to add multiple media files at once, and a “+ Add One” button to add a single file slot to the list.

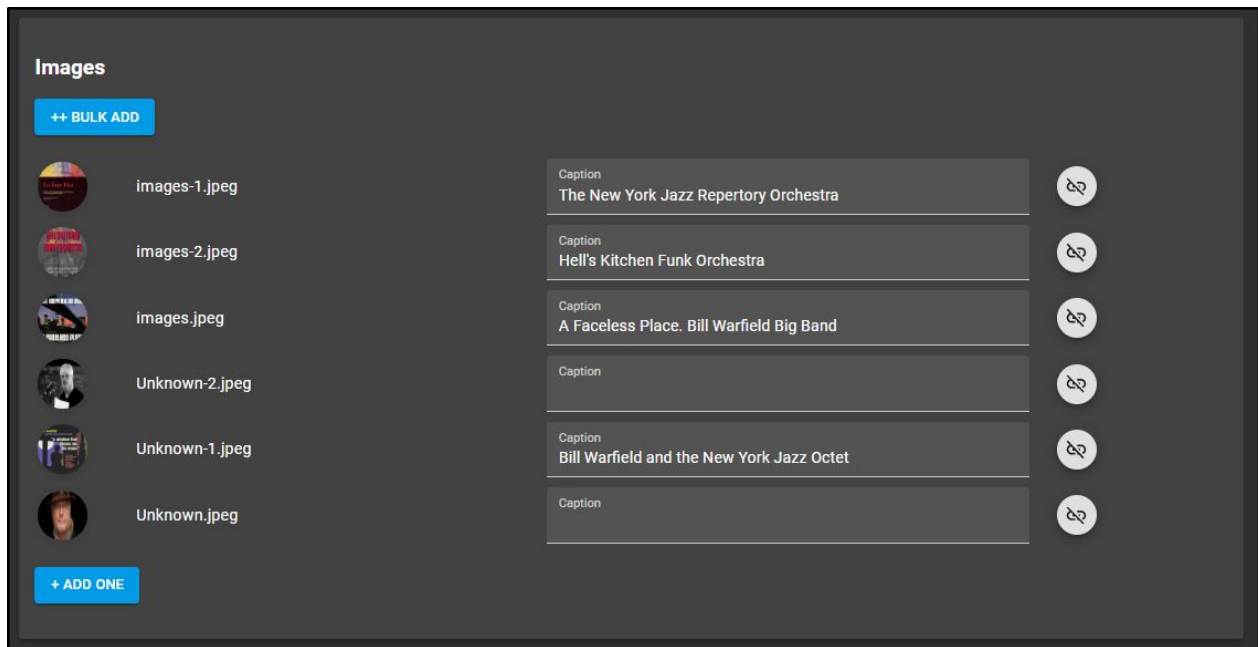


Figure 27. The Images section of the EditContributionView for the Bill Warfield contribution.

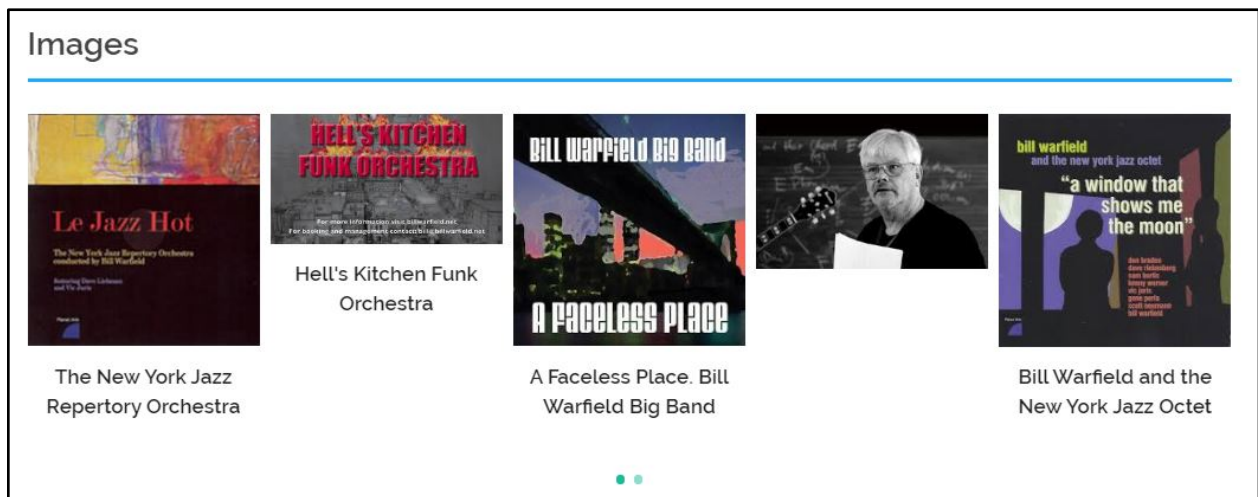


Figure 28. The Images section of Bill Warfield's published page. The portal translates the user's media selections from the previous image into a pre-formatted template. The image carousel rotates automatically.

The Audio section is another instance of MediaUpload in EditContributionView. It transforms a list of audio entries into a list of independent audio players on the page. Each player supports about two lines of text for a caption. However, users may add captions of any length to audio entries, as larger captions will push the rest of the page downward to make space. This section is also optional, and will be omitted if there is no audio.

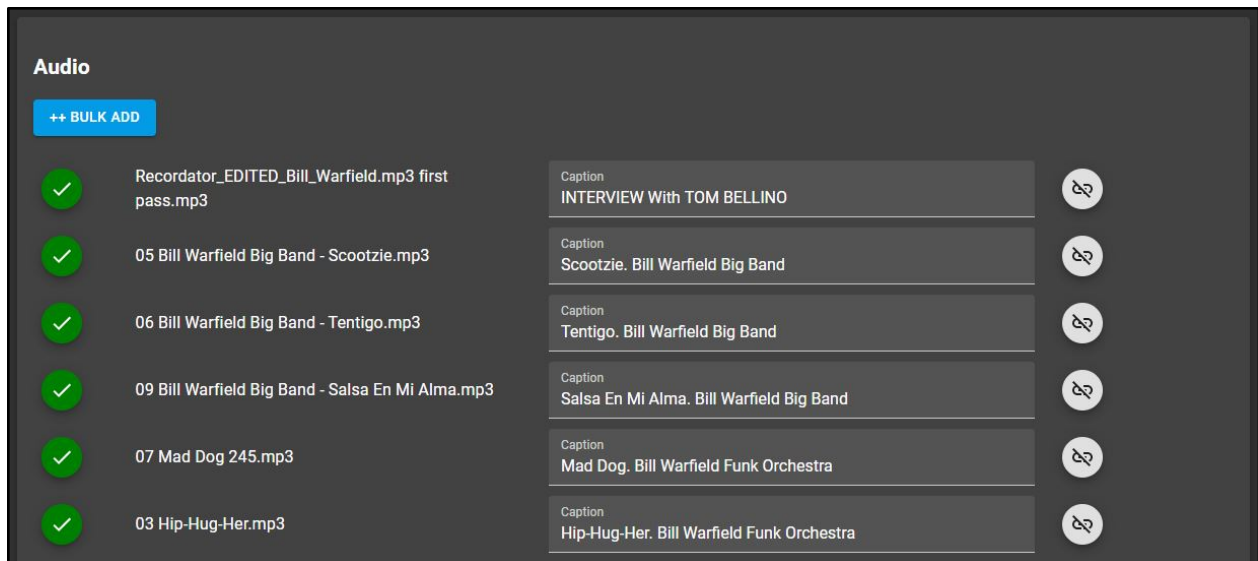


Figure 29. Part of the Audio section of the EditContributionView for the Bill Warfield contribution. There were too many audio files to fit on the image.

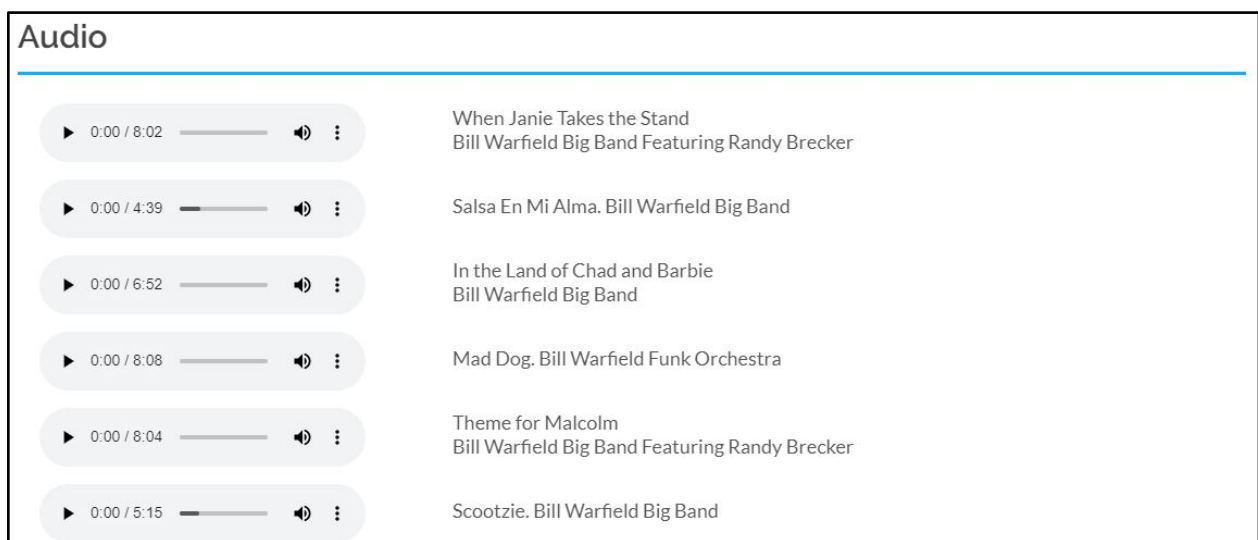


Figure 30. Part of the Audio section of Bill Warfield's published page. The portal translates the user's media selections from the previous image into a pre-formatted template. There were too many audio files to fit on the image.

The Video section is the third and last instance of MediaUpload in EditContributionView. It transforms a list of YouTube links into a list of independent video iframes on the page. Each player supports a caption without size limitations related to spacing issues. This section is also optional, and will be omitted if there is no video.

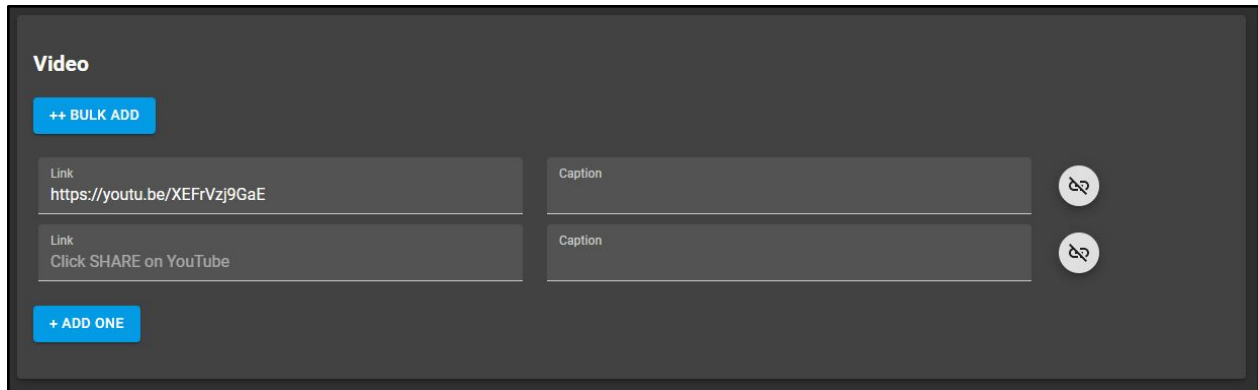


Figure 31. The Video section of the EditContributionView for an example contribution. There were no videos in the Bill Warfield contribution.



Figure 32. The Videos section of an example page. The portal translates the user's media selections from the previous image into a pre-formatted template. There were no videos in the Bill Warfield contribution - this section was omitted in the Bill Warfield contribution.

FileUpload Component

Users select the bio photo and other media on a contribution through the FileUpload helper component. This component represents a media file or link on the page. We consider a FileUpload “empty” when it holds no reference to any file or link. It features four variants, one for each section of the template:

- The Biography variant expects an image and features no caption. The Biography section on EditContributionView contains exactly one FileUpload. As such, clicking the button with a “hide” icon on the right clears the image selection. Users select the bio photo by clicking the floating action button on the left, which displays a thumbnail of the user's selection.



Figure 33. An instance of the FileUpload component in the Bio section, representing the bio photo. No photo has been selected.



Figure 34. An instance of the FileUpload component in the Bio section of Bill Warfield's contribution, representing the bio photo.

- The Images section variant expects an image and allows for a short caption. The component imposes no limit on caption length, but the image carousel on the template may overflow with more than a handful of words. The Images section on EditContributionView is optional and may contain any number of FileUploads. As such, clicking the button with a “hide” icon on the right displays a pop-up warning the user that the image will be removed from the portal (but not from Dropbox or their computer). Clicking the “+ Add One” button at the bottom of the Images section adds an empty FileUpload to the bottom of the list. Clicking the “++ Bulk Add” button adds as many FileUploads as images selected by the user. Users select an image by clicking the floating action button on the left here as well, which similarly displays a thumbnail of the user's selection.

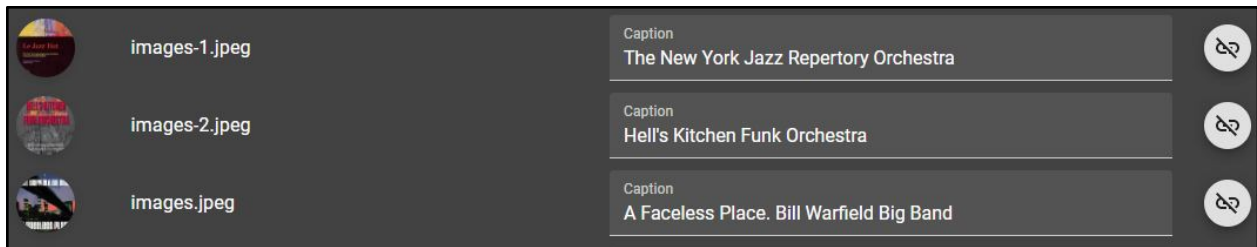


Figure 35. A group of FileUpload components in the Images section of Bill Warfield's contribution.

- The Audio section variant works similar to the Images variant. However, it expects an audio file instead. Additionally, the floating action button on the left of the FileUpload shows a checkmark upon user selection.

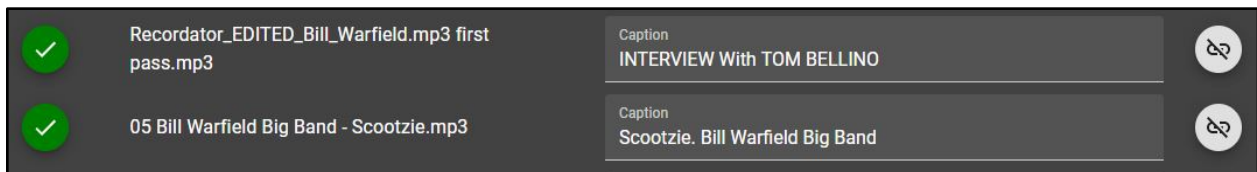


Figure 36. A group of FileUpload components in the Audio section of Bill Warfield's contribution.

- The Video section variant works differently than other variants. It expects a YouTube link through a text box, which replaces the floating action button on the left. If the FileUpload detects a valid YouTube link that is not a “shareable” link (obtained by clicking the “Share” button on YouTube), it converts the link into the proper shareable format once the link text box loses focus.



Figure 37. A group of FileUpload components in the Video section of an example contribution.

The FileUpload left-side buttons and Bulk Add buttons invoke Dropbox's Chooser API. It opens another browser window and navigates to the Dropbox website, with the user's current authenticated Dropbox account. It prompts the user to choose one or more files of a particular category, depending on the FileUpload or Bulk Add button that invoked it. For example, the Bulk Add button in the Images section invokes a Chooser prompting the user to select one or more audio files (any other kind of file is displayed, but disabled and cannot be selected). A FileUpload in the Images section would invoke the Chooser requesting the selection of exactly one image file. Note that even though Dropbox's Chooser allows the user to upload files, portal instructions discourage the user from doing so. Upon selection, the FileUpload or Bulk Add button receives a list of links to user selections, which it uses to fetch thumbnails. Though the Chooser features a convenience Cancel button, users may also close the Chooser window to return to the portal.

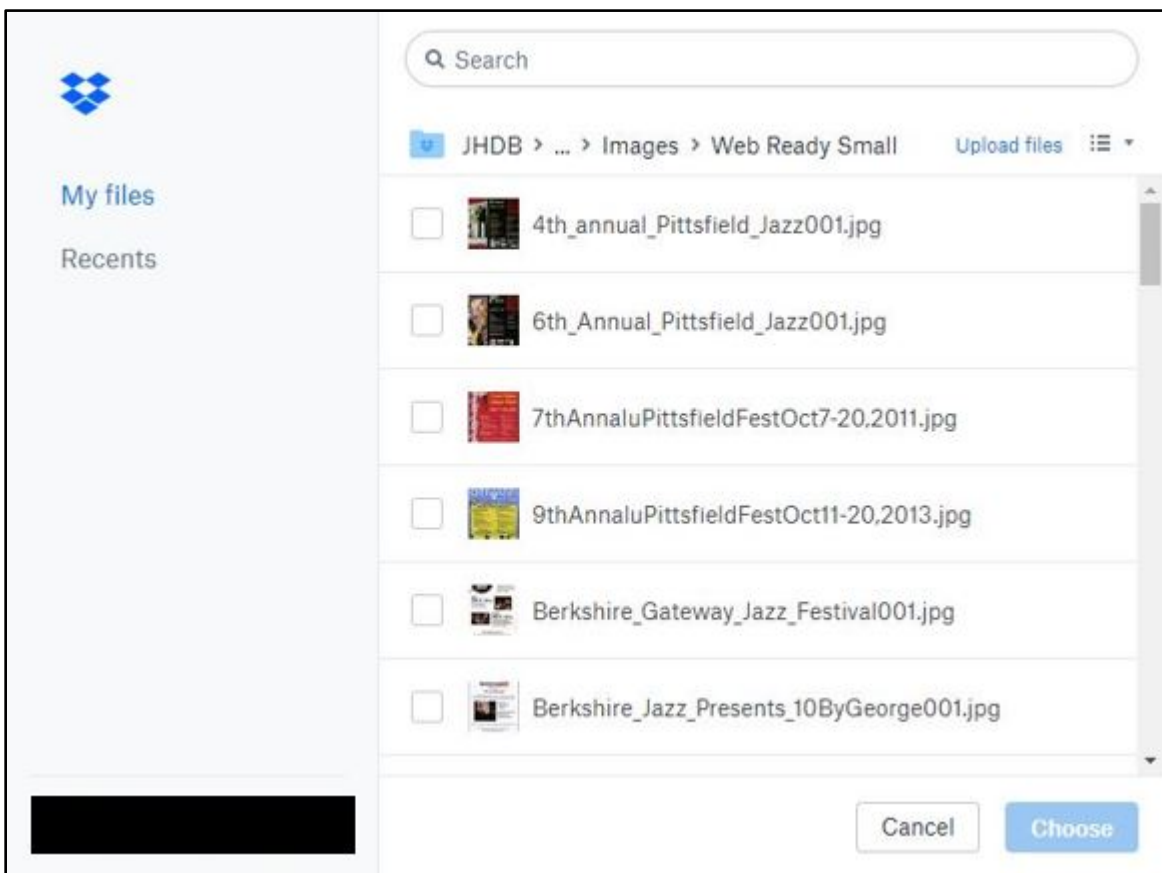


Figure 38. The Dropbox Chooser API website at the time of writing. This Chooser was invoked to select image file types. Sensitive information is redacted on the bottom left-corner of the image.

Preview Tab

Lastly, the secondary tab list under the Pages tab features a Preview tab. Navigating here generates an exact preview of the contribution with its current content, on demand. The only difference between the preview and the published version of a contribution, is that links to other parts of the JHDB site (Biographies, Interviews, Events, etc.) are disabled in the preview. Although the preview is properly

scaled within its boundaries on the portal, the Fullscreen option provides the user a more immersive testing experience, where the preview looks exactly the same on the computer screen as the published version would. Users may exit full screen preview with the browser's back button. Leaving the Preview tab and returning to it requests a fresh preview from the server, regardless of whether changes have been made to the contribution on the portal.

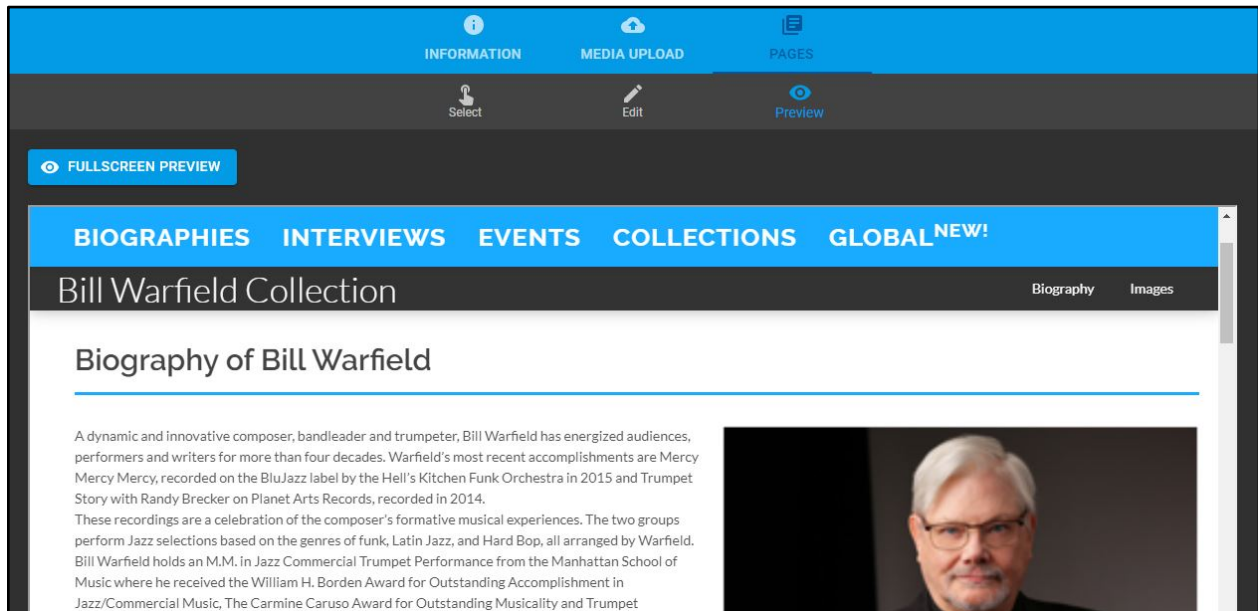


Figure 39. The Preview secondary tab under the Pages tab. It displays a preview of the Bill Warfield contribution.

Evaluating User Experience

Our user experience was evaluated primarily through the use of unmonitored user testing sessions and a Google Forms survey. Due to COVID-19 complications, it was not safe to conduct in-person sessions and we opted not to utilize Zoom or other video calling solutions for monitored interviews. Since some of our target audience consisted of contributors that are not everyday computer users, we did not want the added complexity and technical complications of video conferencing to affect the results of the study.

Study Objective

The objective of this study was to evaluate whether users of the contributor portal find it straightforward and intuitive to log in and add content, as well as identify any confusing or missing aspects of the process to be resolved via further development.

Research Methods

- Unmonitored completion of defined tasks within the contributor portal
- A short online survey assessing the usability and intuitiveness of the tool

Risks

All user testing was conducted online. Participants were instructed on how to access the contributor portal (a self-contained website) with provided login credentials. They were asked to complete a list of simple, non-technical tasks using the contributor portal, and to fill out a short survey upon completion of these tasks. All data were recorded and aggregated anonymously. As such, there was no foreseeable risk for human subjects' safety or privacy.

Study Methodology

Subjects received an email containing an overview of the project, a link to the contributor portal, and Dropbox account credentials with which to access the contributor portal, as well as concise non-technical instructions in written or video format detailing how to fulfill various tasks on the contributor portal. They were also informed of the mission and objectives of the study, and notified of their rights as subjects within the study. Each subject subsequently completed the tasks detailed in the tutorial video and filled out a short survey.

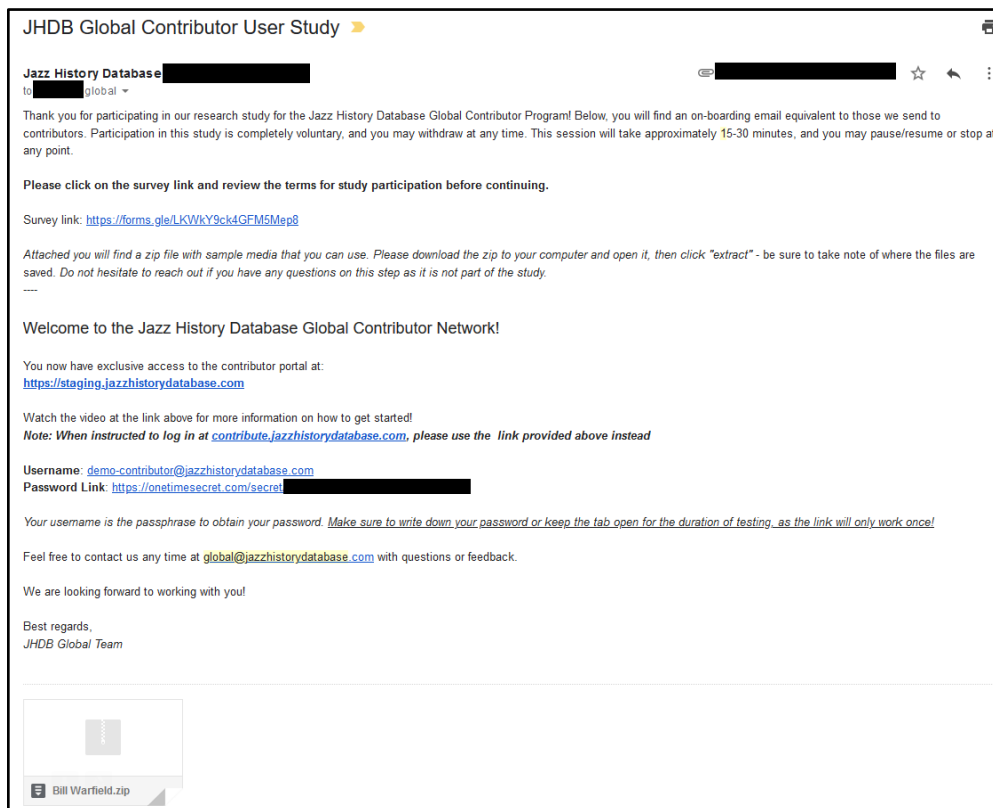


Figure 40. Email sent to study subjects.

Sample Survey Questions

A full copy of the Google Forms survey is included in Appendix B. The questions below are the sample questions submitted to WPI's institutional review board, and are reflective of the questions included in the survey.

- On a scale of 1-5, what is your comfort level and experience with computers? Feel free to elaborate.
- Were you successfully able to log in to the portal? Please detail any difficulties below.
 - [Yes, No, Unsure]
- Were you successfully able to create a new collection? Please detail any difficulties below.
- Were you successfully able to enter a bio, bio photo, and add images to the collection? Please detail any difficulties below.
- Were you successfully able to preview the collection? Please detail any difficulties below.
- Do you feel that the preview page was representative of the content you entered? Please explain in detail.
- Regardless of whether the tasks above were complete, were any of our tasks unintuitive, unclear, or difficult to fulfill through the portal? Please explain in detail.
- Are there any additional things (features, instructions, tutorials) you would like to see on the portal? Please elaborate.
- Do you consent to use of the responses you provided in this form in a research paper for the purposes of evaluating usability and intuitiveness of the tool? All responses will be kept anonymous, and no identifying information will be included.

Obstacles

Scope

Our two biggest obstacles throughout this project were scope, and a series unforeseeable environmental factors affecting our ability to work on the project. While we spent the better part of a year building an understanding of the JHDB's day to day operations and the underlying needs behind the global contributor project, we inevitably discovered new requirements and potential use cases throughout the project. Ideas such as modifying templates within the portal, processing media to ensure web-readiness before publishing pages, and other ideas reflected core pieces of the JHDB's traditional workflow. It was critical for us to continuously identify the highest-value action items to continue making forward progress on our deliverables at the fastest rate possible. Our strategies and workflows for handling scope are further discussed in the [Conclusions & Recommendations](#) section.

Quaran-term

This IQP was completed over the course of D term 2020, spanning the first three months of the COVID-19 lockdowns. Our initial planning for this project had been heavily centered around being on campus, and having direct access to the JHDB lab, as well as face to face contact with stakeholders, contributors and student workers. This quickly changed as testing ramped up and COVID-19 risk in the United States became increasingly prevalent, causing D term to be moved online. We suddenly found ourselves across state lines and unable to utilize on-campus resources we had previously been counting on. Minor obstacles such as having to figure out a new routine, adjust to communicating largely over text, scheduling remote meetings over Zoom, and purchasing budget audio equipment to produce the videos quickly added up to significant delays. While we were lucky to have the resources available to complete

the project remotely, the practical and psychological effects of the quarantine and other current events demanding our attention were definitely prevalent factors in our overall timeline.

Technical Constraints

Some of the technical constraints on the development of the contributor portal itself also slowed us down a bit. One of the more prominent factors in this category was the lack of budget. Designing a fault-tolerant, high-quality piece of software to be reliably accessible on a global scale without incurring any initial infrastructure costs was a considerable challenge to say the least. This required a lot of research into different hosting options for different components of the portal and supporting systems. Understanding the limits of the free tiers and the scalability of each of these options relative to their potential costs proved time consuming. Some aspects of our design, such as the use of a server to allow the user to perform specified actions on the Dropbox administrator account's behalf, were an effective re-engineering of the functionality offered by features such as paid Dropbox team accounts.

Tied into this was the issue of allowing users to upload large files. Since they have to access the portal via a Dropbox Personal account, but all files need to be owned by the JHDB Dropbox Business account, uploading files proved complicated ("Dropbox," 2020). There were numerous security considerations around allowing contributors to upload files as the root administrative dropbox user, and even more considerations when it came to uploading huge files (in some cases multiple gigabyte video files) over potentially slow internet connections. We finally discovered a way to programmatically generate File Upload Requests, which allowed us to generate links to a page where the user could upload files directly to a Dropbox folder of our choosing. File Upload Requests had been used previously to collect digitized media from contributors, and it saved us the headache of having to deal with the low-level network code involved with streaming files.

Outcomes

As previously stated, our goal was to empower developers and remote correspondents of the JHDB with the ability to contribute to the JHDB website remotely, seamlessly, and with little technical skill requirements. Overall, we believe we were successful in accomplishing that goal within the scope of this IQP through the accomplishment of our outlined objectives.

Objectives Achieved

- *The contributor portal [it] must be available online at all times.*

Thanks to careful research, implementation of current technologies, and Site Reliability Engineering standards as detailed in our methodology, the contributor portal is available globally with near-100% uptime. Due to the inherent possibility of unpredictable outages with network-based technologies, we successfully laid out countermeasures to allow us to quickly detect and resolve outages. We received an unexpected opportunity to test these countermeasures on May 1 2020, when the website became unreachable due to a server malfunction shortly after launch. The issue began at 5:15AM EST, and thus took a few hours to be acknowledged. Once the Amplify alarm was acknowledged by our team at 9am however, we were able to quickly diagnose the issue and contact the Academic Research Center at WPI to have the server rebooted. The issue was resolved in under an hour with no negative impact on our users.

- *It must require little familiarity with computers and minimal technical expertise.*

Through our user study, we surveyed a group of users with a variety of experience with computers and the JHDB. While we were unable to survey actual contributors due to COVID-19 related complications, our study group was overall a strong representation of our target audience, with subjects that had some familiarity with computers and varying degrees of familiarity with the JHDB's operations.



Figure 41. Summary of participants' comfort level with computers. Subjects ranged from low-moderate to high levels of comfort with computers, with the majority feeling relatively comfortable. This is a reasonable audience to assess as we are assuming most users of the portal have first-hand or second-hand familiarity with digitizing artifacts.

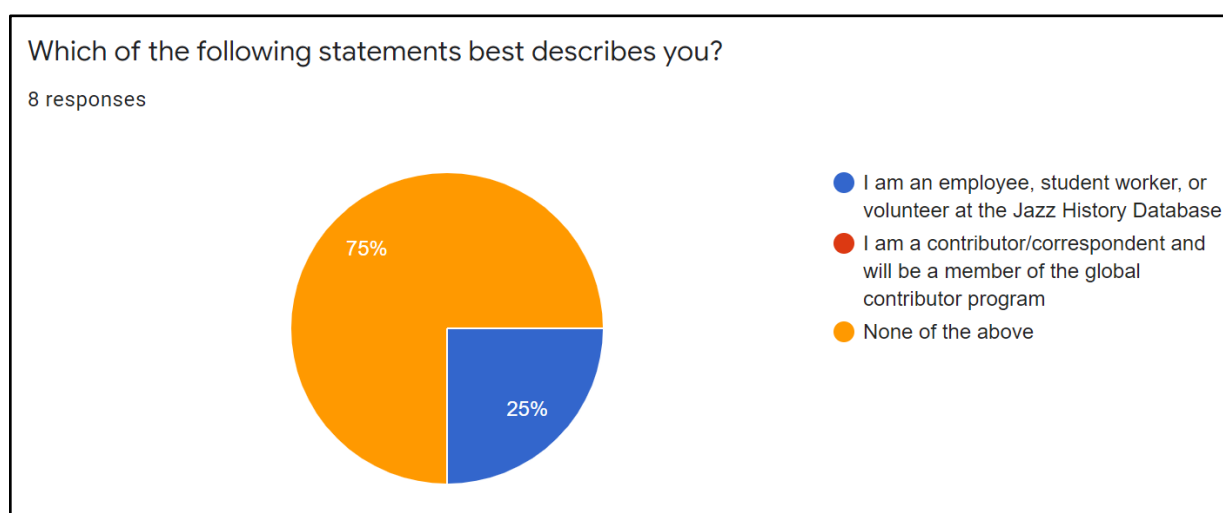


Figure 42. 75% of subjects were generally unfamiliar with the day-to-day operations of the JHDB, with 25% being JHDB workers.

- *It must feature various sources of aid and clarification to the user.*

We successfully developed a professional and thorough video tutorial for the contributor portal. While we did not directly survey users on the quality of the video, several subjects expressed in their comments that they found the video helpful. We received exclusively positive feedback on the video.

- *It must allow the user to upload text, image, audio, and video media into pre-build web page templates.*

This functionality was achieved in full; users can upload images and audio as well as paste text into the portal. All subjects successfully created a page and linked media to it. One subject

expressed that they were unsure whether that step worked, due to a bug that prevented them from previewing the page.

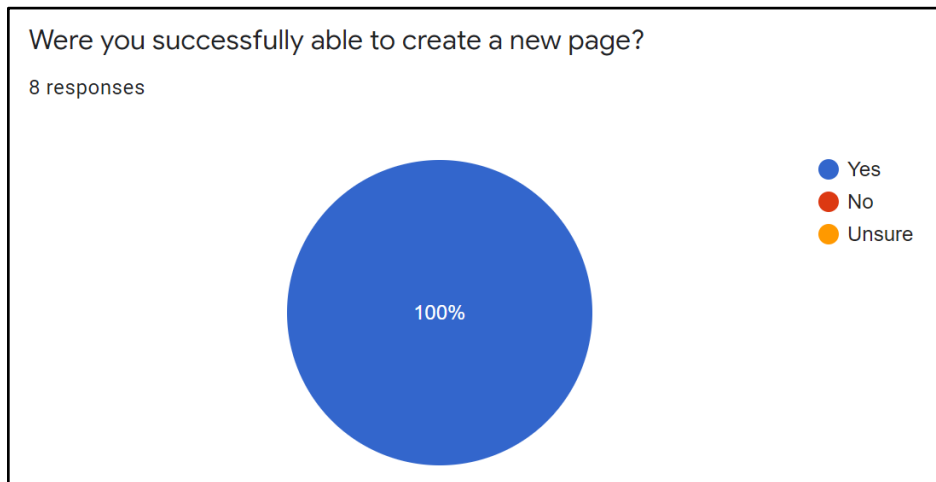


Figure 43. Pie chart of users able to create a page (enlarged).

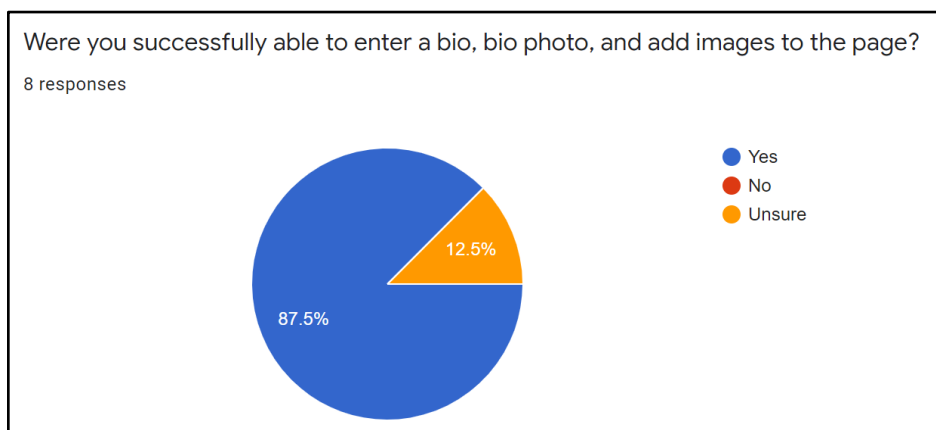


Figure 44. Pie chart of users able to enter bio, bio photo, and images to page (enlarged).

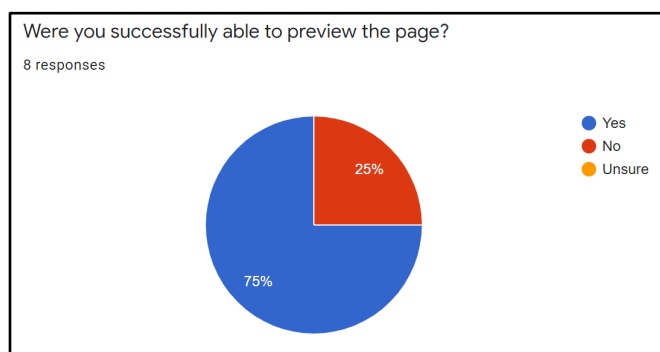


Figure 45. Pie chart of users able to preview the page (enlarged).

2/8 subjects were unable to preview the page to see what it looked like. While not an explicitly defined objective, this falls in line with making the portal intuitive for non-technical audiences and providing aid to the user through the process. We were successfully able to identify the underlying bug in the preview code, and resolve it so that it will not pose an issue in the future ("! PROD RENDERER 502," 2020).

Of the subjects that successfully viewed the preview, all expressed that the preview felt representative of the information they entered into the portal.

- *It must allow the user to request approval to publish a collection of web pages. Once approval is electronically granted by an administrator (such as Professor Falco), the collection shall be published at global.jazzhistorydatabase.com.*

Upon completion of the survey, all user-created pages were marked as “Review Pending,” indicating that subjects had successfully requested review. Professor Falco was successfully able to make revisions and subsequently publish these pages. Some users reached out upon completion of the study to ask, “is that all?” indicating a lack of user feedback for this final step. We plan to address this in the future by giving the user a popup indicating that the page was successfully submitted for review, as is common practice for background tasks that lack inherent user feedback in accordance with the Material Design specification (“Snackbars,” 2020). We additionally hope to send a confirmation email to both the user and reviewer (“Send confirmation,” 2020).

- *It must allow the user to create and maintain multiple collections.*

This feature was achieved as early as our minimum viable product. While not part of the user study, users expressed in the survey comments that they would feel more comfortable building a page the second time around. Through his own usage after the v1 release, Professor Falco quickly and easily produced and published several pages, now live at <https://global.jazzhistorydatabase.com>

- *It must sandbox users to their own particular ‘branch’ of the global JHDB, such that a correspondent will not have access to another correspondent’s collections, unless they express that they work on the same collections.*

This was accomplished in 2 ways. In the user interface code, we filtered pages for normal users such that only theirs would be requested and displayed. For administrators, all pages were visible, but the logged in user’s pages were displayed at the top of the list.

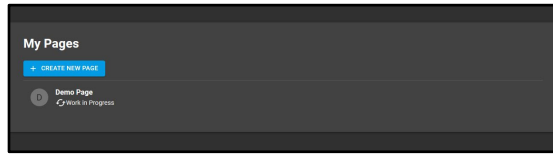


Figure 46.1. Regular users are only able to see their own pages.

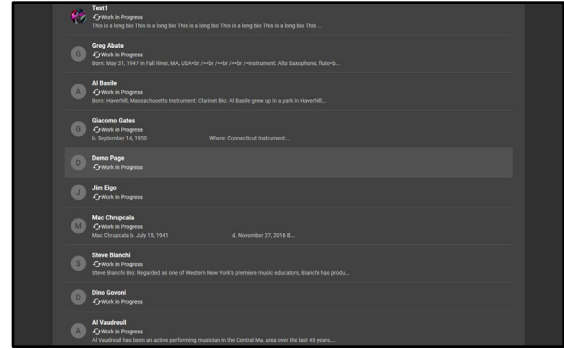


Figure 46.2. Administrators see all pages, but can still see regular user pages listed below their own.

This is additionally enforced by the database rules, which prevent non-administrator users from accessing other users' data, even if they were to modify the browser-side code for the portal.

```
match /Contributions/{document=**} {
  allow create: if userExists() && userIsAuthorized();
  allow read, write: if userExists() && userIsAuthorized() && (userIsAdmin() || userIsOwner(resource));
}
```

Figure 47. Firestore rule preventing non-administrator users from accessing other users' data.

- *Its source code and repositories must include extensive documentation, to facilitate onboarding future developers into the project.*

The code for the contributor portal is open source, and is publicly available on GitHub at <http://github.com/jazzhistorydatabase/jhdb-cms>. The on-boarding process is thoroughly documented in a README file in accordance with industry standard practices. A copy of the README file is included in the Appendix D. Server configurations are also documented there, but are accessible by invitation only to mitigate exposing any potential security vulnerabilities.

- *Lastly, the portal must protect the integrity and identity of the JHDB and its members, such that no personally identifiable information is unintentionally publicized, and no remote correspondent could accidentally damage the portal or the website.*

As mentioned above, users' access to data is enforced both in the contributor portal user interface (which runs on the user's computer within their web browser), as well as server-side by the Firestore data access rules. Any routes in our server software (such as the publish route, which requires access to the JHDB Dropbox Business administrator account to upload published pages), also verify the identity of the user via the Dropbox OAuth API and Firebase Admin SDK before performing any actions ("OAuth guide", n.d.; "Verify," 2020). Users only interact with the contributor portal through JHDB-issued accounts, so the portal does not store any user contact information. This also allows us to immediately reset the password of these accounts in the event

that one is compromised to protect the user's own data. Since regular users do not have access to publish their own pages without administrator approval, even a malicious hacker with the user's login credentials cannot affect published content. Activity is fully logged, so any potentially malicious activity can be detected. To protect privacy, names of users are omitted from the logs, and all user actions are logged using unique identifiers for their dropbox account, which our team can manually cross-reference with our user record in the Firebase console to know which user to contact should their account be compromised, or to help troubleshoot bugs reported by a particular user.

```
POST 200 215 B 1.3 s Chrome 83 /publish
2020-06-07T17:10:25.481Z User request publish for page yacine-boulares
2020-06-07T17:10:25.482Z Successfully validated token for user account_id: dbid: [REDACTED] requesting to publish yacine-boul
2020-06-07T17:10:25.526Z Successfully get admin users, checking account_id: dbid: [REDACTED]
2020-06-07T17:10:25.526Z User dbid: [REDACTED] is admin, beginning publish
2020-06-07T17:10:25.528Z User request preview for collection: yacine-boulares
2020-06-07T17:10:25.528Z Fetching contribution by name: "yacine-boulares" -> "Yacine Boulares"
2020-06-07T17:10:25.577Z Found 1 matches for name: yacine-boulares
2020-06-07T17:10:25.615Z Successfully fetched 6 audio entries for Yacine Boulares doc id: [REDACTED]
2020-06-07T17:10:25.622Z Successfully fetched 12 images for Yacine Boulares doc id: [REDACTED]
2020-06-07T17:10:25.623Z Successfully fetched 2 videos for Yacine Boulares doc id: [REDACTED]
2020-06-07T17:10:25.623Z Rendering preview template with name: Yacine Boulares found doc id: [REDACTED]
2020-06-07T17:10:25.623Z Successfully rendered preview template with name: Yacine Boulares doc id: [REDACTED]
2020-06-07T17:10:25.628Z Got preview render for page yacine-boulares, uploading to dropbox
2020-06-07T17:10:26.742Z Successfully uploaded html file to dropbox for yacine-boulares
```

Figure 48. Sample logs for publish endpoint. User and data identifiers have been redacted.

Deliverables

Contributor Portal

Our primary deliverable was the contributor portal itself, which is online at <https://contributor.jazzhistorydatabase.com>. If anyone reading this paper would like to learn more about the JHDB and the contributor portal, they can learn more at <https://global.jazzhistorydatabase.com> or reach out to the team directly via email at global@jazzhistorydatabase.com.

Tutorial Video

The instructional video explaining the use of the portal was our second key deliverable. This video is public at <https://contributor.jazzhistorydatabase.com/> such that prospective contributors can view the video and familiarize themselves with the portal before even receiving their login credentials. The video runs about 15 minutes, and provides a well-paced step-by-step walkthrough, starting with obtaining their login credentials, logging into the portal, all the way through building and publishing their first page.

Trailer Video/Virtual Research Showcase

Since International Jazz Day (April 30th) fell within the timeframe of our IQP, we also created a general informational video on the JHDB and the Global Contributor Project. This trailer video is posted on the frontpage of <https://global.jazzhistorydatabase.com>. A variant of this video was submitted for WPI's Virtual Research Showcase, which included additional background and historical context for the IQP specific to a WPI audience. This video, along with a Powerpoint presentation containing a summary of the video's content, are both available on the Virtual Research Showcase website: <https://virtualresearchshowcase.wpi.edu/dept-pages/interactive-media-and-game-development.html#123>

Technical Documentation

Our last deliverable was thorough documentation of the process and technical documentation on the portal itself for future maintainers and volunteers at the JHDB. While our team intends to remain engaged with the JHDB to help maintain and improve the contributor portal for the foreseeable future, the goal of this documentation was to allow us to hand the project off entirely in the future, to someone with the required technical knowledge to maintain it in our stead. The procedural parts of this documentation were provided in the form of the aforementioned videos and this paper. Technical documentation is provided alongside the code on GitHub ("Jazzhistorydatabase/jhdb-cms," n.d.). A copy of the on-boarding documentation for development of the portal is included in the Appendix D.

Conclusions & Recommendations

While this project focused primarily on the design, creation, and documentation of the aforementioned deliverables, it was an amazing learning experience. Our team successfully worked with a stakeholder to deliver a valuable product, and balance the technical and non-technical needs of the project to guarantee the usability of said product. The core project was completed in eight weeks - an extremely ambitious timeline for a project with such a vast scope. It is worth noting that both members of the core team had prior familiarity with the Jazz History Database as well as the technical skills required to complete the project, and had accumulated a lot of the background knowledge required for the project over the course of a year. That said, the process was not without its hiccups and we provide our concise reflections below, in hopes that they will prove useful to future project teams.

Scope is Key

This is a very difficult factor to account for, especially when working with an established organization to deliver a product based on the “vision” of multiple individuals. Our stakeholders (primarily Professor Falco and Professor Drnek) had many amazing and exciting ideas about what the contributor portal could be, as well as differing understandings and assumptions of the technical implementation and work involved in making those a reality. Had we opted to design around all of these ideas (or even a majority of them), we would have blown ourselves dramatically out of scope and this paper would have read more like a post-mortem than a description of our process and deliverables.

As we highlighted in the [Obstacles](#) section, it is critical to understand the needs of the project as thoroughly as possible, and to identify the highest value action items at any given time. This practice is not limited to the planning phase of a project - constantly and repeatedly evaluating, prioritizing, and down-scoping is always necessary to deliver a usable product by a deadline. For others pursuing this kind of project in the future, we highly recommend Jeff Sutherland’s book *Scrum: The Art of Doing Twice the Work in Half the Time* (Sutherland, 2016). The principles of Scrum are a standard across many design and engineering disciplines, and the strategies discussed in the book were extremely valuable to us in gracefully coping with the sudden obstacles posed by the series of unexpected hurdles of Spring 2020.

Focus on Users

When designing a tool for end-users, the biggest pitfall that enthusiastic engineering students usually succumb to is focusing on technical specifications. Functionality is important - if a software tool does not include the functionality users need, it is useless. However, functionality is not necessarily proportional to the value of the resulting product to the end user. This is the mistake that doomed previous attempts at the same project we have documented in this paper. Understanding the needs of stakeholders and addressing their ideas is important, yet paradoxically, stakeholders often don’t entirely understand what they want (specifically, they don’t have a working knowledge of the technical constraints at play, or they may be unaware of other technologies or options that may solve their problem in a different way from what they have envisioned).

To anyone looking to pursue a similar project: it is key to ask a lot of questions and build an understanding of the core problem facing the users of your tool. The solution may not line up one-to-one

with their suggestions, but it is important to understand the context you may be lacking that informs those suggestions. Most importantly, always remember that your tool will be used by a human being, with different background, experiences, and expectations from you. A step or detail that seems obvious to you may be thoroughly confusing to your tool's target user. Likewise, the user's expectations may not be immediately obvious or even logical in the context of your own knowledge and experience. Make sure to deliver tangible prototypes early that test your assumptions about the user's expectations, and ask questions to contextualize the user's feedback on what you show them. Focusing on the end-user of a product is always critical to developing a tool that will be not only functional, but intuitive and even enjoyable.

Future Work

While the scope of this project is completed, there is a lot of future work remaining for the contributor portal. There are various minor bugs and potential optimizations to address, as well as additional use cases for the portal to encompass (adding more templates, or even a template creator into the portal, linking pages together, etc). We took on this project because we recognized the unique value of the JHDB, and their critical need for this technology to allow their services to remain available to the global jazz community. Our development work on the project is ongoing, and open-source contributors/developers (both from fellow WPI students/faculty and external assistance) are greatly welcome. The state and future work of the JHDB Global Contributor Project is maintained alongside the code for the Global Contributor Portal at <https://github.com/jazzhistorydatabase/jhdb-cms/projects>.

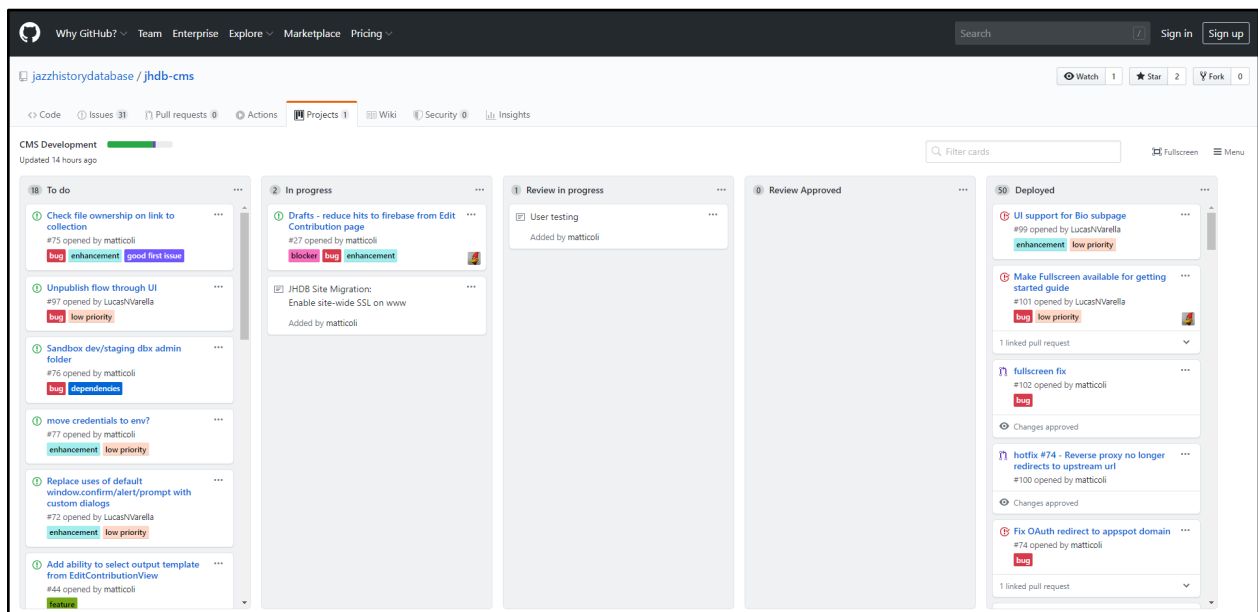


Figure 49. Active project board on the contributor portal GitHub Repository.

References

References

! PROD RENDERER 502 error on preview: Dino govoni #109. (2020, Jun 5,).

<https://github.com/jazzhistorydatabase/jhdb-cms/issues/109>

2018-2019 inquiry seminars and practicums in humanities and arts. (2018). Worcester Polytechnic

Institute. <https://www.wpi.edu/sites/default/files/inline->

[image/Inquiry%20Seminar%20Practicums%202018-2019_6.pdf](https://www.wpi.edu/sites/default/files/inline-image/Inquiry%20Seminar%20Practicums%202018-2019_6.pdf)

Add ability to select output template from EditContributionView #44. (2020, Jun 29,).

<https://github.com/jazzhistorydatabase/jhdb-cms/issues/44>

Amanda carr collection. Jazz History Database.

<http://www.jazzhistorydatabase.com/archives/lorraine-feather/>

The app engine standard environment. (2020). Google Cloud.

<https://cloud.google.com/appengine/docs/standard>

Cloud firestore. (2020). Firebase. <https://firebase.google.com/docs/firestore>

Dark theme. Material Design. <https://material.io/design/color/dark-theme.html>

Docker | empowering app development for developers. Docker.com. <https://www.docker.com/>

Dropbox plan comparison. <https://www.dropbox.com/business/plans-comparison>

Hafiz, S. (). Why are PHP function signatures so inconsistent? [closed].

<https://softwareengineering.stackexchange.com/questions/32541/why-are-php-function-signatures-so-inconsistent>

Jazz history database. Jazz History Database. <http://www.jazzhistorydatabase.com/index.php>

Jazzhistorydatabase/jhdb-cms - GitHub. GitHub. <https://github.com/jazzhistorydatabase/jhdb-cms>

Lorraine feather collection. Jazz History Database.

<http://www.jazzhistorydatabase.com/archives/lorraine-feather/>

Material design. Material Design. <https://material.io/design>

Material-UI: A popular react UI framework. Material-UI. <https://material-ui.com/>

Matticoli, M. (2020). *Jhdb-cms/cloudbuild.yaml*. GitHub.

<https://github.com/jazzhistorydatabase/jhdb-cms/blob/master/cloudbuild.yaml>

Matticoli, M., & Varella, L. (2020). *Jhdb-cms/firestore.rules*. GitHub.

<https://github.com/jazzhistorydatabase/jhdb-cms/blob/master/firestore.rules>

Naming developer environments. (2020). Google Cloud.

<https://cloud.google.com/appengine/docs/standard/php/creating-separate-dev-environments>

Norman, D. (2013). *The design of everyday things* (Rev. and expanded ed. ed.). Basic Books.

OAuth guide. Dropbox. <https://www.dropbox.com/developers/reference/oauth-guide>

Portainer management, docker user interface, container software. Portainer.io.

<https://www.portainer.io/>

Send confirmation email on review requested #122. (2020, Jun 7,).

<https://github.com/jazzhistorydatabase/jhdb-cms/issues/122>

Snackbars. Material Design. <https://material.io/components/snackbars>

Steve cancelli. Jazz History Database.

http://www.jazzhistorydatabase.com/content/musicians/cancelli_steve/

Steve cancelli - photo gallery. Jazz History Database.

http://www.jazzhistorydatabase.com/content/musicians/cancelli_steve/gallery4.php

Steven cancelli obituary - worcester, massachusetts. (2016). Legacy.com.

<https://www.legacy.com/obituaries/name/steven-cancelli-obituary?pid=182802749>

Sutherland, J. (2016). *Scrum - the art of doing twice the work in half the time*.

Verify ID tokens. (2020). Firebase. <https://firebase.google.com/docs/auth/admin/verify-id-tokens>

Video conferencing, web conferencing, webinars, screen sharing - zoom. Zoom Video

Communications. <https://zoom.us/>

WordPress. WordPress. <https://wordpress.org/>

Appendix A: Sample User-Study Invitation

We include here a sample email sent to a participant of our user study. All participants were individually sent the same email from global@jazzhistorydatabase.com, with a few minor differences. Each participant received a different one-time password link, and some participants were provided with an account called testcontributor@jazzhistorydatabase.com instead.

Participant contact information and password link were redacted in the following sample email.

Email Invitation

From: Jazz History Database <global@jazzhistorydatabase.com>
Sent: Saturday, May 30, 2020 7:23 PM
To: [redacted]
Cc: global@jazzhistorydatabase.com
Subject: [EXT] JHDB Global Contributor User Study

Thank you for participating in our research study for the Jazz History Database Global Contributor Program! Below, you will find an on-boarding email equivalent to those we send to contributors. Participation in this study is completely voluntary, and you may withdraw at any time. This session will take approximately 15-30 minutes, and you may pause/resume or stop at any point.

Please click on the survey link and review the terms for study participation before continuing.

Survey link: <https://forms.gle/LKWkY9ck4GFM5Mep8>

Attached you will find a zip file with sample media that you can use. Please download the zip to your computer and open it, then click "extract" - be sure to take note of where the files are saved. Do not hesitate to reach out if you have any questions on this step as it is not part of the study.

Welcome to the Jazz History Database Global Contributor Network!

You now have exclusive access to the contributor portal at:
<https://staging.jazzhistorydatabase.com>

Watch the video at the link above for more information on how to get started!

Note: When instructed to log in at contribute.jazzhistorydatabase.com, please use the link provided above instead

Username: demo-contributor@jazzhistorydatabase.com
Password Link: [redacted]

Your username is the passphrase to obtain your password. Make sure to write down your password or keep the tab open for the duration of testing, as the link will only work once!

Feel free to contact us any time at global@jazzhistorydatabase.com with questions or feedback.

We are looking forward to working with you!

Best regards,
JHDB Global Team

Appendix B: Survey Questions

Provided here is a complete documentation of our user study survey. This survey was conducted using Google Forms.

Jazz History Database Contributor Portal Survey

The objective of this study is to evaluate whether users of the contributor portal find it straightforward and intuitive to log in and add content, as well as identify any confusing or missing aspects of the process to be resolved via further development.

Participation in this study is completely voluntary. You may stop using the contributor portal at any time by closing the corresponding tab in your web browser. Once you are done, we ask that you fill out this short survey detailing your experience using the tool, though this is entirely voluntary as well and you may choose not to do so. This feedback will be used in our research paper to evaluate whether or not we have succeeded in creating an intuitive and usable tool, as well as to help improve the tool for future users. No identifying information will be recorded or included in the paper, and if you do not wish for your feedback to be included in the paper you may indicate that in this survey, or choose not to fill it out. If at any point you wish to withdraw from this study, simply reply to our original email saying "I no longer wish to participate in this study" - you may choose whether or not to provide your reason for doing so, though that information will not be used as part of the study.

This session will take approximately 15-30 minutes, and you may pause or stop at any time. Please return to the on-boarding email message to continue prior to filling out this survey.

On a scale of 1-5, what is your comfort level and experience with computers?

	1	2	3	4	5	
Least	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Most

Figure 50. Survey question 1.

Which of the following statements best describes you?

- ☐ I am an employee, student worker, or volunteer at the Jazz History Database
- ☐ I am a contributor/correspondent and will be a member of the global contributor program
- ☐ None of the above

Figure 51. Survey question 2.

Feel free to elaborate on your comfort level and experience with computers.

Your answer

Figure 52. Survey question 3.

Were you successfully able to log in to the portal?

- ☐ Yes
- ☐ No
- ☐ Unsure

Figure 53. Survey question 4.

Please detail any difficulties logging into the portal.

Your answer

Figure 54. Survey question 5.

Were you successfully able to create a new page?

☐ Yes

☐ No

☐ Unsure

Figure 55. Survey question 6.

Please detail any difficulties creating a page below.

Your answer

Figure 56. Survey question 7.

Were you successfully able to enter a bio, bio photo, and add images to the page?

☐ Yes

☐ No

☐ Unsure

Figure 57. Survey question 8.

Please detail any difficulties you encountered while adding information to your page below.

Your answer

Figure 58. Survey question 9.

Were you successfully able to preview the page?

☐ Yes

☐ No

☐ Unsure

Figure 59. Survey question 10.

Please detail any difficulties previewing the page below.

Your answer

Figure 60. Survey question 11.

Do you feel that the preview page was representative of the content you entered? Please explain in detail.

Your answer

Figure 61. Survey question 12.

Regardless of whether the tasks above were complete, were any of our tasks unintuitive, unclear, or difficult to fulfill through the portal? Please explain in detail.

Your answer

Figure 62. Survey question 13.

Are there any additional things (features, instructions, tutorials) you would like to see on the portal? Please elaborate.

Your answer

Figure 63. Survey question 14.

Do you consent to use of your responses to the above questions in a research paper for evaluating the usability and intuitiveness of the tool? All responses will be kept anonymous, and no identifying information will be included.

☐ I consent to use of the responses I provided in this form.

Figure 64. Survey question 15.

Appendix C: Survey Results

Provided here are the summary results of the survey as provided by Google Forms, for all questions included in our user study survey. Individual survey responses were omitted to preserve the anonymity of our participants. Please refer to Appendix B to view the structure of questions from the participant's perspective.

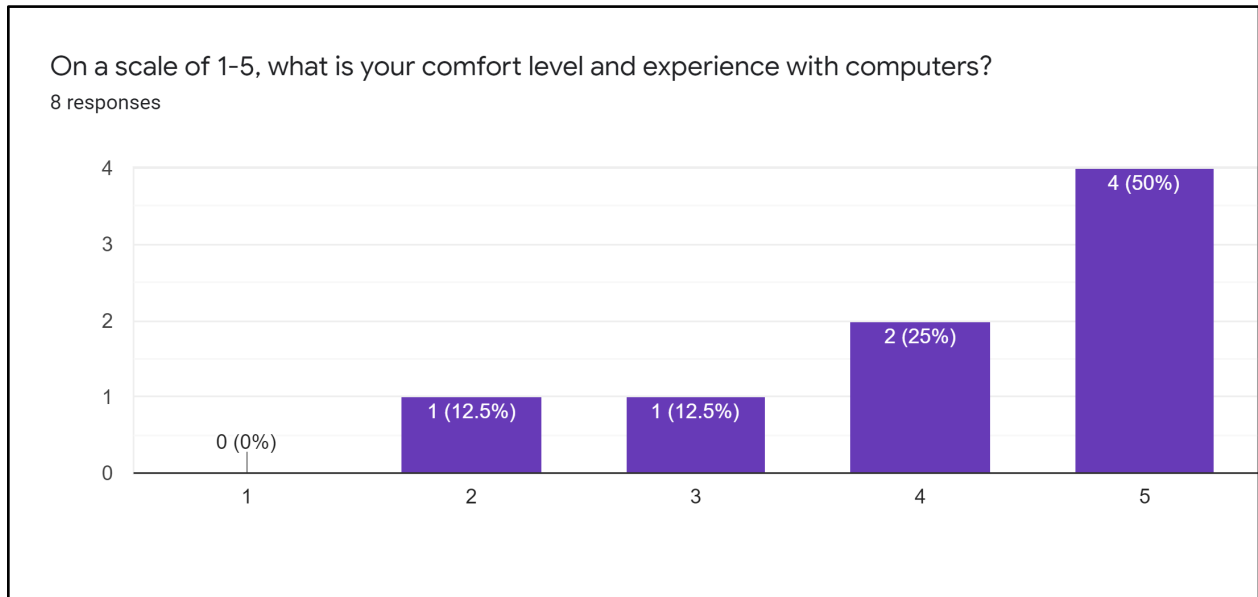


Figure 65. Summary of results for question 1.

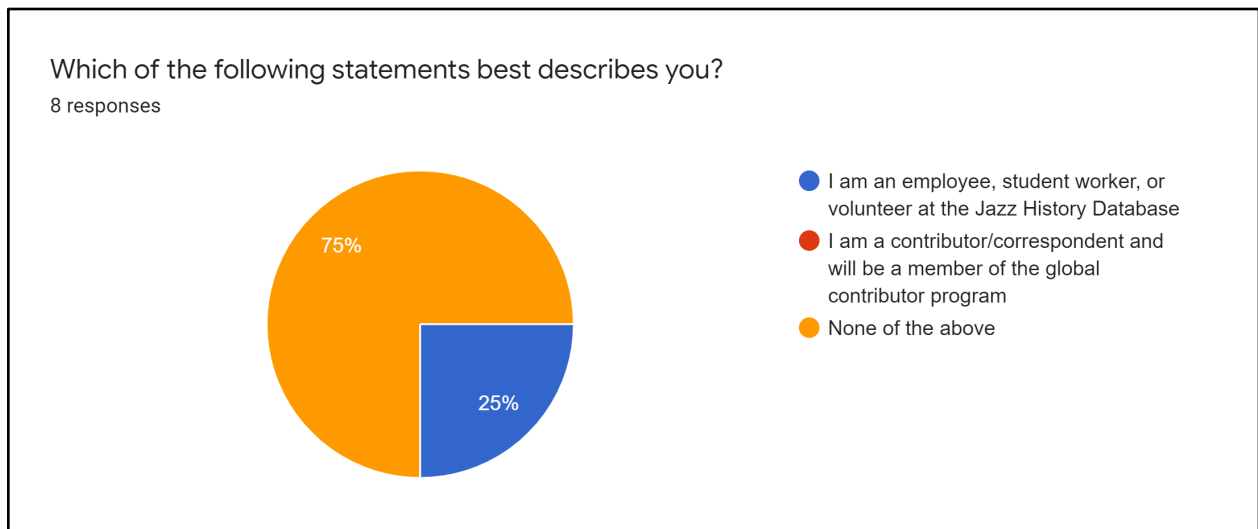


Figure 66. Summary of results for question 2.

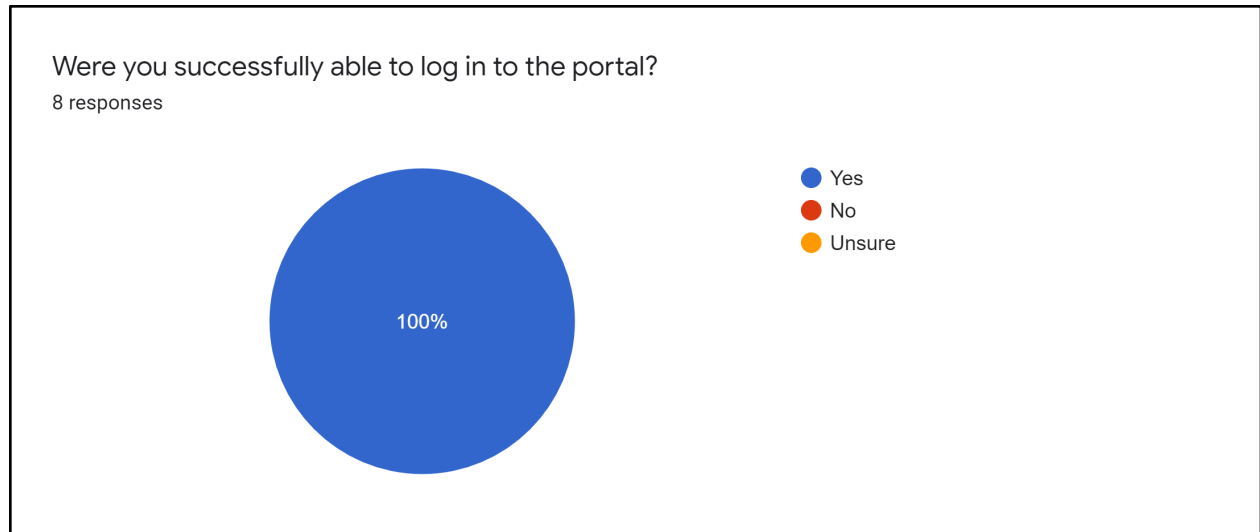


Figure 67. Summary of results for question 4.

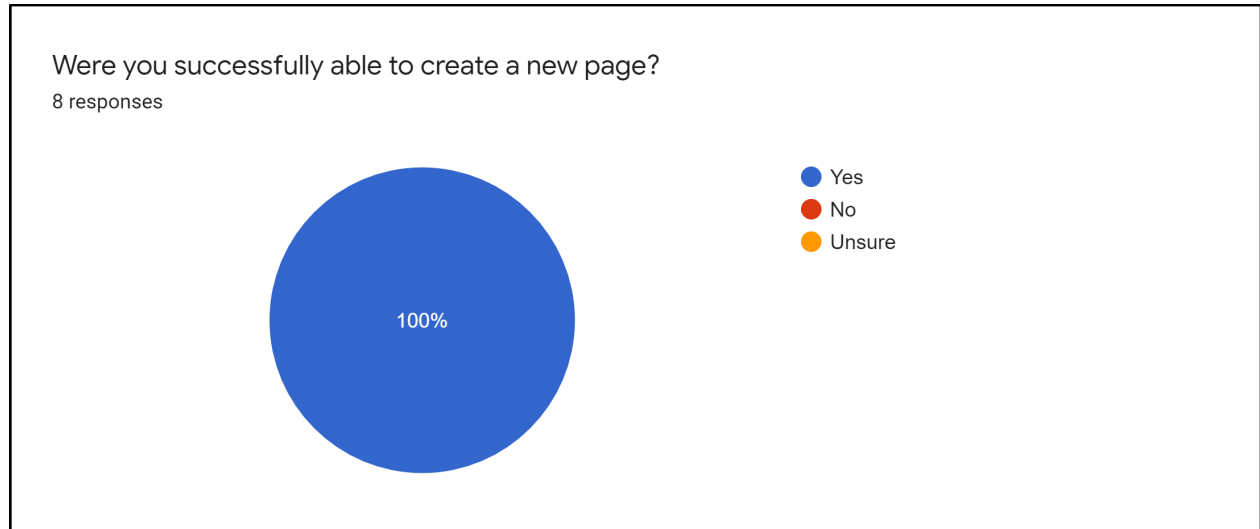


Figure 68. Summary of results for question 6.

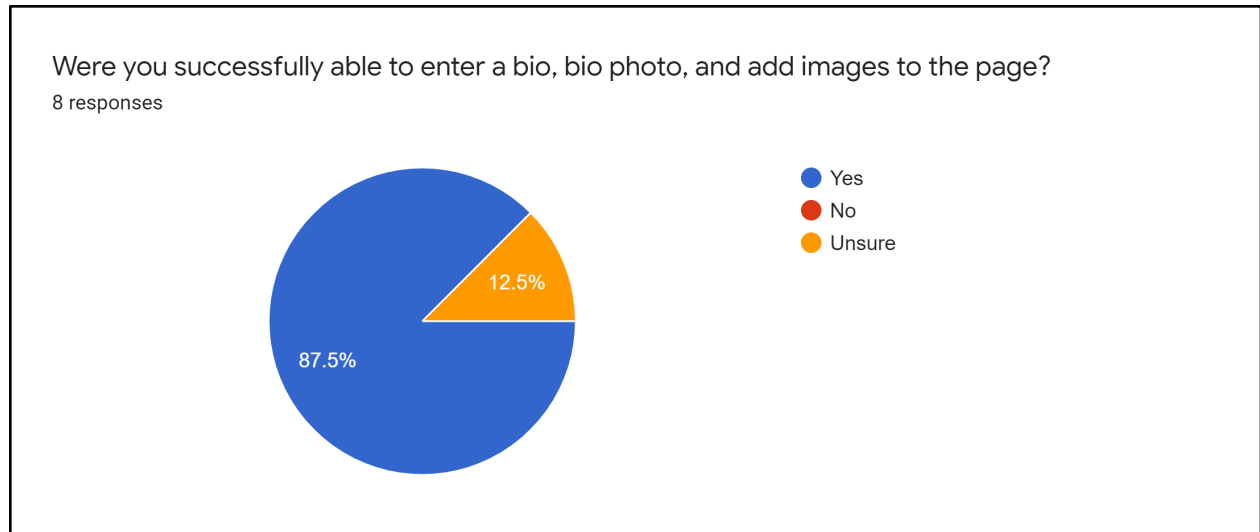


Figure 69. Summary of results for question 8.

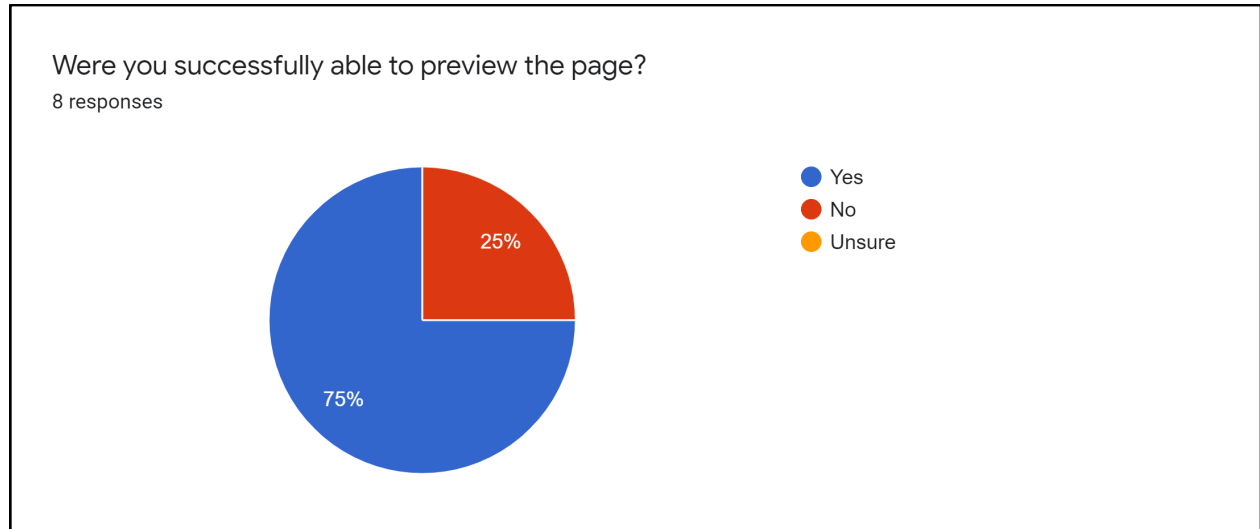


Figure 70. Summary of results for question 10.

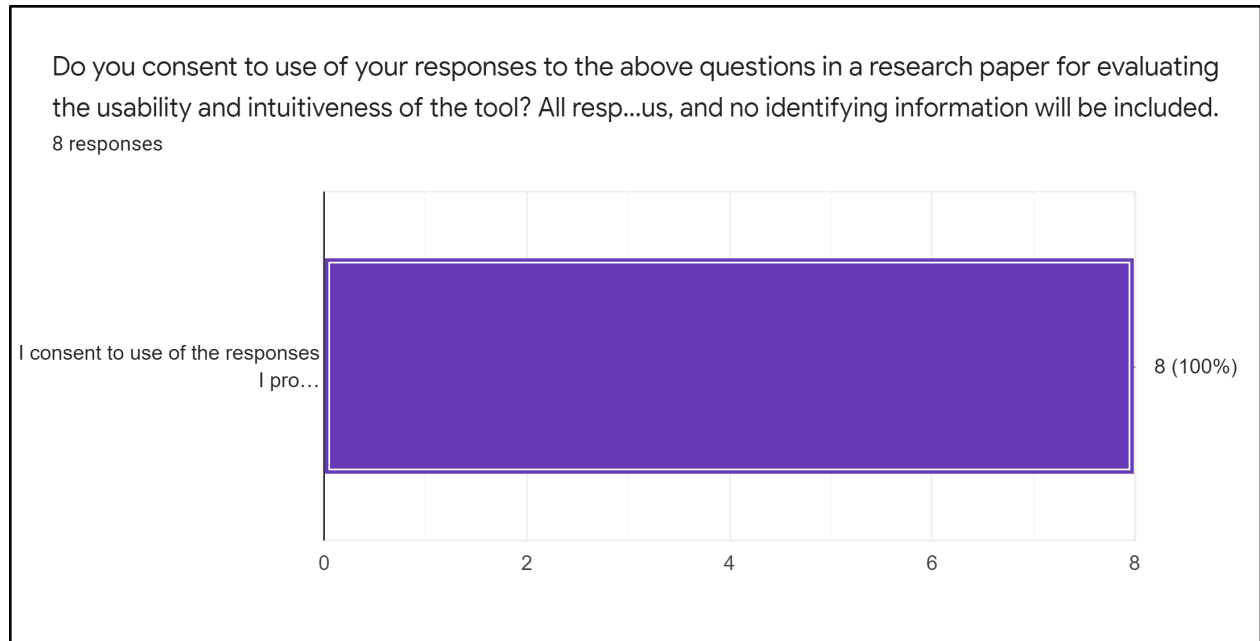


Figure 71. Summary of results for question 15.

Appendix D: On-Boarding Documentation

README.md

Jazz History Database Content Management System

This project was bootstrapped with [Create React App](#).

Getting Started

You can clone the project by running

```
git clone https://github.com/jazzhistorydatabase/jhdb-cms.git
```

NOTE: Running this app requires unchecked credentials files for the server and client. (Development access to Google Cloud is required in order to download using the links below)

- **Server credentials:** rename to `server-creds.json` and place in project root
- **Client credentials:** rename to `client-creds.json` and place in `src/`

To run dev server using Docker:

```
docker-compose up
```

To run dev server using npm:

```
npm install
npm run dev
```

Available Scripts

In the project directory, you can run:

```
npm install
```

Installs all dependencies

```
npm run dev
```

Runs the app in the development mode.
Open <http://localhost:8080> to view it in the browser.

The page should reload if you make edits.
You will also see any lint errors in the console.

```
npm test
```

Launches the test runner in the interactive watch mode.
See the section about [running tests](#) for more information.

npm run build

Builds the app for production to the `build` folder.

It correctly bundles React in production mode and optimizes the build for the best performance.

The build is minified and the filenames include the hashes.

Your app is ready to be deployed!

npm start

Runs app in production mode. This will serve the content in the `build` folder (must build first). To apply code changes, rebuild.

Packaging

This project utilizes two bundling systems to minimize the bundle size being pushed to GCP:

- [Backpack by jaredpalmer](#) is used to bundle and minify server code in production builds. This allows the server to run with a few files containing only active code, as opposed to using `node_modules/`, which causes issues with App Engine's file cap.
- [Parcel](#) is used for client-side bundling for local dev and production. In dev, the parcel watcher transpiles the contents of `client/` faster than a react dev server. In prod, an optimized bundle is built and outputted to the same `dist` directory, so dev and prod servers serve client code the same way.

Workflow

- Tasks are tracked via issues - if you come across a bug or would like to request a feature, please submit it in the [issues tab](#)
 - Kanban boards for the active and past development cycle reside in the [projects tab](#)
 - Development is tracked in the [#db-dev channel on the Jazz History Database Slack](#) (accessible to WPI students and by invitation)
-

Production Environment

Google Cloud Platform

This application is hosted on Google Cloud Platform and Firebase. Permissioned users can deploy, troubleshoot, and otherwise manage production services at <https://console.cloud.google.com/home/dashboard?project=jhdb-cms>

Firebase

Data and authentication is handled by firebase, and can be managed via the Firebase console at <https://console.firebase.google.com/u/0/project/jhdb-cms/>

There are also Staging and Dev Firebase Instances

- Staging: <https://console.firebase.google.com/u/0/project/jhdb-staging/>
- Development: <https://console.firebase.google.com/u/0/project/jhdb-dev/>

Automatic Deployment

All commits to the `master` branch are auto-deployed to the [production environment](#) on GCP's App Engine. This is the correct workflow for regular releases.

Manual Deployment

Authenticated users with the Google Cloud SDK installed can run `gcloud builds submit --config cloudbuild.yaml .`
Alternatively, to deploy to staging: `gcloud builds submit --config cloudbuild-staging.yaml .` This should be used in extreme circumstances only (i.e. in case of service outage)! All releases should be peer reviewed and auto-deployed via pull request.

Learn More

You can learn more in the [Create React App documentation](#).

To learn React, check out the [React documentation](#).

Code Splitting

This section has moved here: <https://facebook.github.io/create-react-app/docs/code-splitting>

Analyzing the Bundle Size

This section has moved here: <https://facebook.github.io/create-react-app/docs/analyzing-the-bundle-size>

Making a Progressive Web App

This section has moved here: <https://facebook.github.io/create-react-app/docs/making-a-progressive-web-app>

Advanced Configuration

This section has moved here: <https://facebook.github.io/create-react-app/docs/advanced-configuration>

Deployment

This section has moved here: <https://facebook.github.io/create-react-app/docs/deployment>

`npm run build` fails to minify

This section has moved here: <https://facebook.github.io/create-react-app/docs/troubleshooting#npm-run-build-fails-to-minify>