Drug Label Explorer

Spring 2022

Software Engineering Capstone, CSCI E-599 Section 2

Group Members

Ken Brown, Agi Kajanaku, Leo Landau, Sam Negassi, Ky Nguyen

Customer

David Edelen

Teaching Staff

Peter Henstock, Roman Burdakov

Table of Contents

Drug Label Explorer: An Exploratory Tool Utilizing Information Extraction to Analyze FDA and EMA Drug Labels

ABSTRACT

The primary purpose of a drug label is to provide relevant information to healthcare providers regarding the dispensing and administration of medication. The information contained on drug labels can be a valuable source of information or a jumping-off point for a variety of industries and research specialties. These labels have the potential to provide patients with more than just information; by comparing extracted data, new perspectives on adverse reactions, drug classification, drug interactions, and precision medicine can be gained in a variety of ways. Drug Label Explorer is an exploratory tool that includes robust search functionality that supports a wide variety of queries, such as data filtering and aggregation using multiple attributes via searchable queries, as well as label comparison between FDA(US) and EMA versions (EU). This process begins by extracting raw data from each database and parsing it into various formats. The FDA uses XML files, which are well-known for their structure, whereas the EMA uses HTML/PDF files, which have a less structured format. MySQL Full-Text Search was used to match user-supplied queries via the innoDB engine for the search feature. The version comparison view compares two or three dissimilar drug labels, either within or across drug classifications, by listing each section's text and highlighting text changes. DLE accessed the previously described data sources and extracted 46,005 FDA-approved drug labels and 1,284 EMA-approved drug labels. In the case of FDA sections, we iterated from over 950 section titles to 83 section titles by grouping more specific similar groups under a single generalized title. We demonstrate the effects of these features by displaying interactive views on the software's front-end. To meet user needs, we developed software that dynamically extracts data from multiple sources, including the FDA/DailyMed, the European Medicines Agency, and user-uploaded labels. The DLE can then filter user-input semantic or exact search queries and compare labels based on section title and content.

Keywords: Drug labels, FDA, EMA, information extraction, regulatory information

INTRODUCTION

Drug labels contain critical information on any prescription or over-the-counter medication under sections such as generic name, dosage, administration, clinical pharmacology, boxed warning, indication, and pharmacokinetics among other

drug-relevant information [17, 18, 19]. In this paper, ‘drug labeling’ is used as an umbrella term to encapsulate all the information in the structured drug labels of both the FDA and EMA approved labels. The main purpose of a drug label is to inform healthcare providers of relevant information on dispensing and administering medication in order to assist patients mitigate drug-related medical errors and other serious adverse reactions. Additionally, access to this information can help make more informed decisions if a patient is on several medications and needs to ensure that a particular combination is appropriate for their needs [30]. However, drug labels have the ability to provide more than just information to patients, by comparing extracted information, new perspectives can be gained in several ways for adverse reactions, drug classification, drug interactions, and precision medicine.

Drug label sections such as Boxed Warning, Warnings and Precautions, and Adverse Reactions allude to drug-related adverse events, with vast application to pharmacovigilance and
drug safety research in order to improve compliance, reporting, and signal interpretations [25]. While using a standard language to describe adverse events is not required, most descriptions of adverse events utilize a standard vocabulary, which makes studying adverse event data in drug labels more effective. Drug classification can be defined in a variety of ways depending on the use, including clinical trials, mechanistic studies, and chemical structure. The classifications used in medicine labeling include Chemical Ingredients, Established Pharmacologic Class, Mode of Action, and Physiologic Effect. These classification schemes make it easy to evaluate a drug's classification during the review process and justify changing its labeling [31].

The findings relating to drug–drug interactions and their associated adverse events in drug application are summarized in a section of the drug labeling. Specific inquiries, such as which HIV medicines are known to interact with methadone and which pharmaceuticals will interact with disulfiram are then queried against the labeling data [31]. Drug labels are known to include a significant amount of pharmacogenomics biomarkers. These biomarkers are anticipated to influence the efficacy and adverse effects of the medications in patients from specific population subgroups [29]. This data aids in the detection of new trends and the frequency of genetic variability linked to increased public health hazards.

As evident, information on drug labels can be a great source or starting point for a plethora of industries and research specialities. A prevalent challenge is that most resources available utilize an explanatory research approach to how their users can navigate their tools. While there are many benefits to explanatory tools, users must know exactly what information is to be found. Exploratory tools have the benefit of

discovery where data can be extrapolated in a way that permits users to gain deeper insight into analysis. Currently, users who want to utilize drug labeling to further research interests must navigate tools like DailyMed, FDALabel and EMEA databases to gather raw data of all approved drug labels available in the US and EU. Researchers investigate other sources to hypothesize potential drug labels of interest, store those labels in a third party such as a local machine and either manually compare and contrast differences or implement general-use analytical tools. To account for version control of labels, users must keep a pulse on updates through staying current in relative research or directly monitoring the database systems.

DRUG LABEL EXPLORER

Drug Label Explorer (DLE) is an exploratory tool that provides capabilities to robust search functionality that supports a wide variety of queries, including data filtering and aggregation using several different attributes. This allows the user to identify and compare desired labels across agencies on an international level which no existing solution has provided before as a publicly available resource. In addition to this, DLE has a unique feature that permits the database to continuously update itself as databases in use update themselves for all agencies in use. This is clearly indicated to users who would be able to view all saved versions through search results and note respective changes. This feature has been shown as highly useful to initial users as approximately 450 drug label updates are published by the FDA every week [4]. These updates can originate from a variety of sources such as spontaneous reports (52%), clinical trials (16%), and pharmacokinetic studies (11%) [3].

Users who want to create their own labels based on the structure and language of other drug labels have the ability to upload custom drug labels privately to their accounts and be able to access those privately or share directly from DLE to others as they prefer. This feature is especially useful to drug manufacturers to optimize unpublished drug labels by comparing their labels with the rest of the database available on our website. To mitigate tedious processes, users can save queries and compare desired search results all on the DLE platform without having to outsource storing labels of interest to a different system and can choose to compare labels from their saved searches or from search results as needed.

DLE alleviates pain points of users who rely on current solutions. Integrating good design principles is a key way to ensure usability, and positive experiences using our product. The User Interface (UI) uses high contrast coloring to account for readability

and accessibility. Negative space is used meticulously to visually guide the user throughout their process by taking advantage of the Law of Proximity. Flexibility of DLE features such as advanced search and filters. The navigation is highly intuitive and decluttered to support users in a good familiar experience [34].

RELATED WORK

For the US, the FDALabel [16] and DailyMed [17] databases, as well as the EMA [18] database for the European Union, are the principal sources of approved prescription pharmaceutical labels. These web-based platforms include basic search capabilities in addition to raw data from all approved drug labels. The FDA maintains a database of pharmaceutical Safety-related Labeling Changes (SrLC) that enables users to search for drug label changes, but it lacks the key capability of searching for specific text inside a drug label document. However, the data is only provided in XML or PDF format, and there is no way to search for specific sections or compare medication labels. Additionally, the user interfaces are not intuitive. We also looked at RxList [20], ReedTech [21], WizMed [22], the Cerner Website [23], and Drugs.com [24].

These commercially available websites host searchable databases of medicine label information that differ in terms of content, search options, and accessibility. Some of these commercial databases are subscription-based [21, 22], while others are ad-supported [20, 24], and still others are restricted to healthcare enterprise use only [23]. The majority of these commercially available websites assert that the FDA provided the data for their databases, although others claim to have data from the EMA and other nations, with one claiming to have data from nine countries [23]. While we were unable to verify the functionality of the fee-based and private sites, the ad-supported sites' capacity to analyze the content of drug labels fell short of enabling us to quickly locate the drug labels that matched our query. RxList.com and Drugs.com, for example, allow users to search the text of medicine labels. However, they do not support "exact match" searches, and it is occasionally unclear why the labels are included in the search results. Additionally, these services do not make it easy to compare the content of two drug labels in order to identify comparable text.

DATA SOURCES

FOOD AND DRUG ADMINISTRATION (DAILYMED)

The Food and Drug Administration (FDA is a centralized US government agency charged with guaranteeing the safety, efficacy, and security of human pharmaceuticals, biological products, and medical devices[19]. DailyMed is the authorized source for FDA label information, which is frequently referred to as package inserts. It is a freely accessible medication labeling database resource that the National Library of Medicine (NLM) submits to the FDA and contains the most recent versions of drug labeling submitted to the FDA. It is the authorized source for FDA labeling information. These labels are defined in the Health Level Seven (HL7) Structured Product Labeling (SPL) standard, which defines the different sections of a medicine label [28]. It utilizes Logical Observation Identifiers Names and Codes (LOINC) to connect the various components and subsections of human prescription medicine and biological product labeling. The FDA requires that medicine labels be "informative and accurate, without being promotional in tone or deceptive in content." In the United States, there are about 51,000 human prescription medications and biological products approved for use [25], and that figure is rapidly increasing by a few hundred new approvals each year.

EUROPEAN MEDICINES AGENCY

The European Medicines Agency (EMA) is a decentralized agency that is responsible for the scientific evaluation, inspection, and safety monitoring of pharmaceuticals in the European Union, Iceland, Norway, and Liechtenstein [18]. The EMA provides information about medications available to the public. Their database contains information about many stages of a drug's life cycle, encompassing early development, first evaluation, and post-approval status (withdrawals and updates), along with safety reviews.

METHODS

The workflow used to collect the drug labels from FDA and EMA sources can be broken down into four general steps. FDA utilizes XML files which are known to be highly structured, while EMA uses HTML/PDF files and has a less structured format thus making it more challenging in respect to the more sporadic nature of the structure. To ensure accuracy in how data is managed, a process was facilitated to convert the formats of both FDA and EMA to allow for more efficient and optimal comparisons. This process is initialized with extracting the raw data from each database and parsing the extracted data in different formats. For FDA drug labels, the NDC code was used to successfully map each drug label based on the labeler code section which is assigned by the FDA to a manufacturer, the product code which gets

assigned to the drug product by the manufacturer, and the package code which distinguishes package configurational changes.

Information extraction was used to siphon information from the raw data, this was done especially for the less structured formats. The method of pre-processing drug labels involves preparing the text for processing using computational linguistics tools such as sentence splitting, tokenization, and morphological analysis. The process of detecting and categorizing concepts entails detecting and classifying references to persons, things, locations, and events, as well as other pre-specified types of concepts. The objective of connecting concepts is to establish linkages between the retrieved concepts. Unifying is the process of presenting collected material in a standardized format. Eliminating noise entails removing redundant data. Enriching your knowledge base refers to the process of ingesting extracted knowledge into your database for future usage[32].

To ensure accuracy, DLE employs an identification scheme that links directly to the FDA/EMA product information for each unique drug label. This provides users with transparency and the ability to flag any potential errors for a more positive experience. Due to the fact that the formats of some of these FDA drug labels vary in terms of similar section titles with slight wording variations, DLE relied on domain experts to validate condensing available section names to bound sections describing the same type of pertinent information about the particular drug. This was done to make navigation easier for users and to improve search results by encompassing drug information under a broader term.

For the search feature, MySQL Full-Text Search was used to match user-supplied queries via the innoDB engine; this was chosen to allow users to explore terms that are not necessarily identical to those in the DLE database while maintaining faster results than other search indexes. When user inputs do not precisely match database terms, DLE uses natural language full-text search mode to determine the contextual meaning of the queried search. To accomplish this, our product used Natural Language Processing (NLP) to parse the search query and segment the phrase or sentence into components that could be used to differentiate keywords from other terms. This is done to reduce word redundancy and account for words with multiple meanings in order to optimize meaning[33].

The individual drug label page lists all existing versions of a drug label. From this page, users can select any two versions of the label and launch a comparison view. The version comparison view lists each section text of the two versions side by side and highlights the text changes (the text *diff*) between the two versions. DLE uses a

python library named *diff-match-patch* to compare the section texts and highlight the differences between the two versions of the drug label. A drop-down filtering menu allows the user to filter the results to only show the sections with different texts or show only sections with matching texts. Alternatively, a user can also filter down to the comparison result of a specific section. In addition to the version comparison, users have the ability to compare two or three dissimilar drug labels, within or across drug classification. To do this, users can select two or three dissimilar drug labels from the search result page and display them side by side in order to compare them. This allows for comparing text across drug labels authored by different manufactures and within a certain classification. Unlike the version comparison view, the *labels comparison* view does not highlight differences in the text. This is an intentional design decision because dissimilar drug labels are expected to have very different texts and highlighting those differences will only be a noise. The labels comparison page displays each section's text side by side and provides similar filtering features discussed above.

RESULTS

DLE accessed the previously described data sources and extracted 46,005 drug labels from the FDA, of which 9,202 were not yet approved and 36,803 were active labels. DLE was able to extract data from 1,284 drug labels from the EMA database. 6.7 percent of the sections in these EMA files did not map to their anticipated section. To address this, the sections containing these mapping errors were omitted while maintaining the remainder of the files for a larger corpus and ensuring the accuracy of the remaining features. Once parsed the multiple data sources, it was discovered that nine sections of the EMA's drug labels accounted for an average of 93 percent of the label sections. These frequently encountered sections include Indications, Posology, Contraindications, Warnings, Interactions, Pregnancy, Driving Effects, and Overdose. In the case of FDA sections, we iterated from over 950 section titles to 83 section titles by grouping more specific similar groups under a generalized title per grouping. These 83 sections contain 96 percent of all information on drug labels, with the remaining 4% being saved in a "other" section.

The results for DLE are primarily feature-based; we developed data parsing strategies that mitigate risks associated with the acquisition of additional drug label data while maintaining high levels of accuracy. We accomplished this by simplifying the extraction process and ensuring that both new data and high levels of accuracy are acquired.

Figure 1 illustrates the outcomes of these features by displaying views on the software's front-end with which the user can interact. **Fig1.1** demonstrates how the landing page doubles as a form for users to submit their search terms. **Fig1.2** expresses an example of potential results for an exact search user query, where items can be selected for comparison and the compare button directs the user to the next step. **Fig1.3** highlights the view for comparing two labels based on their section content, and users can also compare three search results in the same view if necessary. **Fig1.4** captures the readability of a single drug label, which can be accessed directly or via comparison features. **Fig1.5** includes a simple, easy-to-use upload custom label feature that will save the label to the DLE database and display it privately to the user beneath the submit button in the My Labels section.

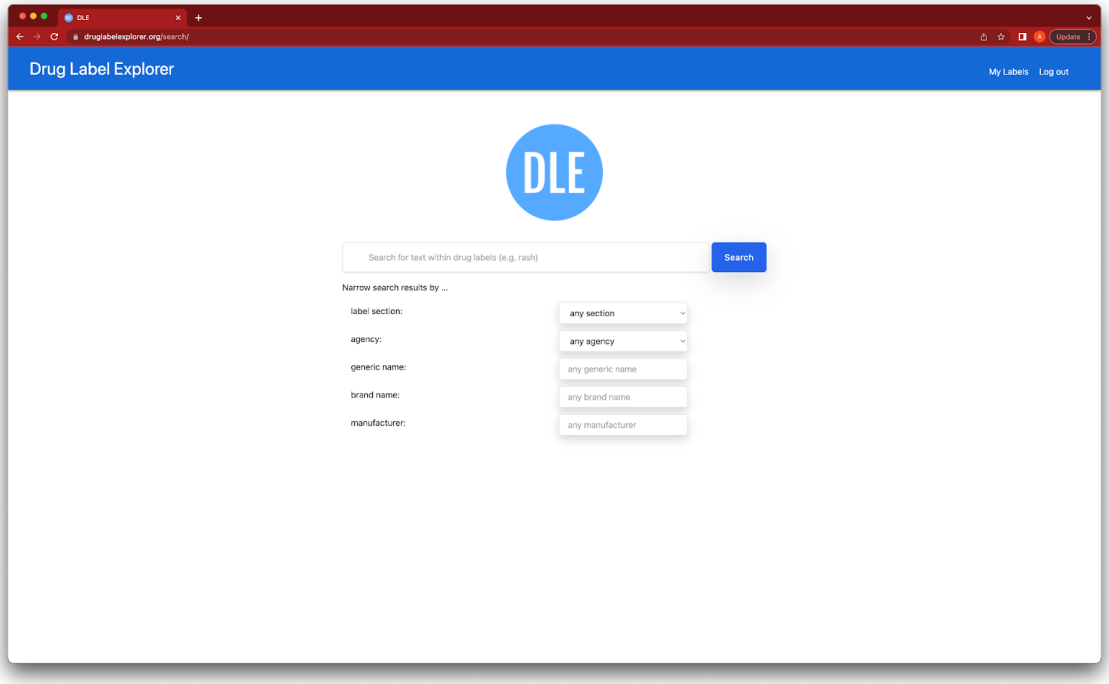


Figure 1.1: Drug Label Explorer Landing Page View

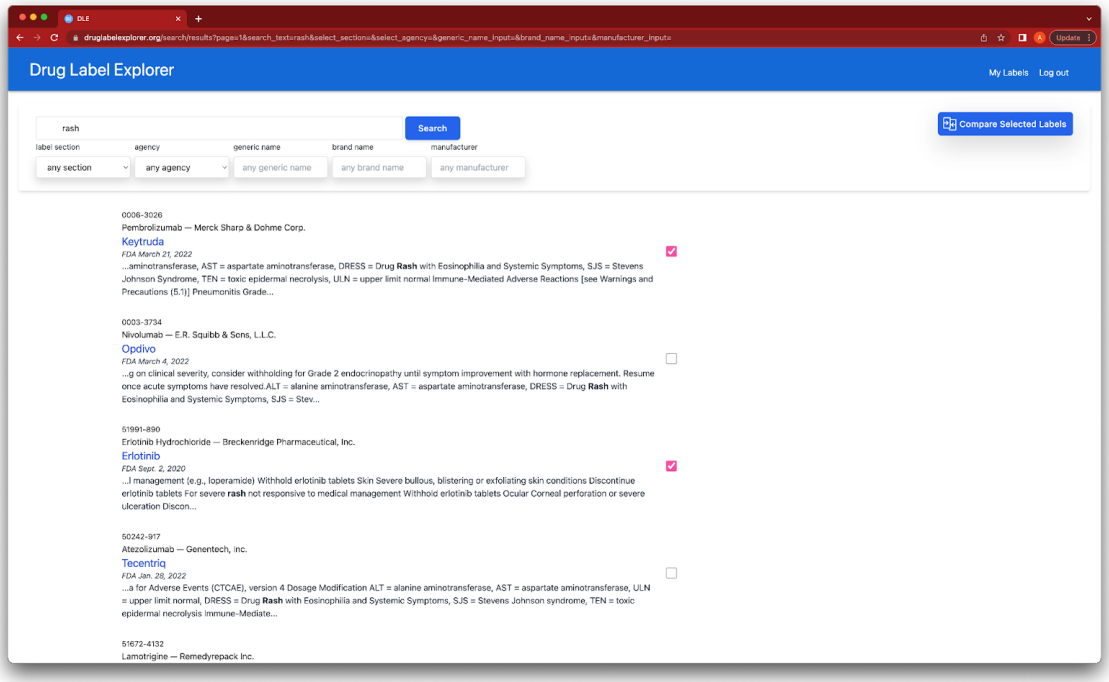


Figure 1.2: Drug Label Explorer Search Results View

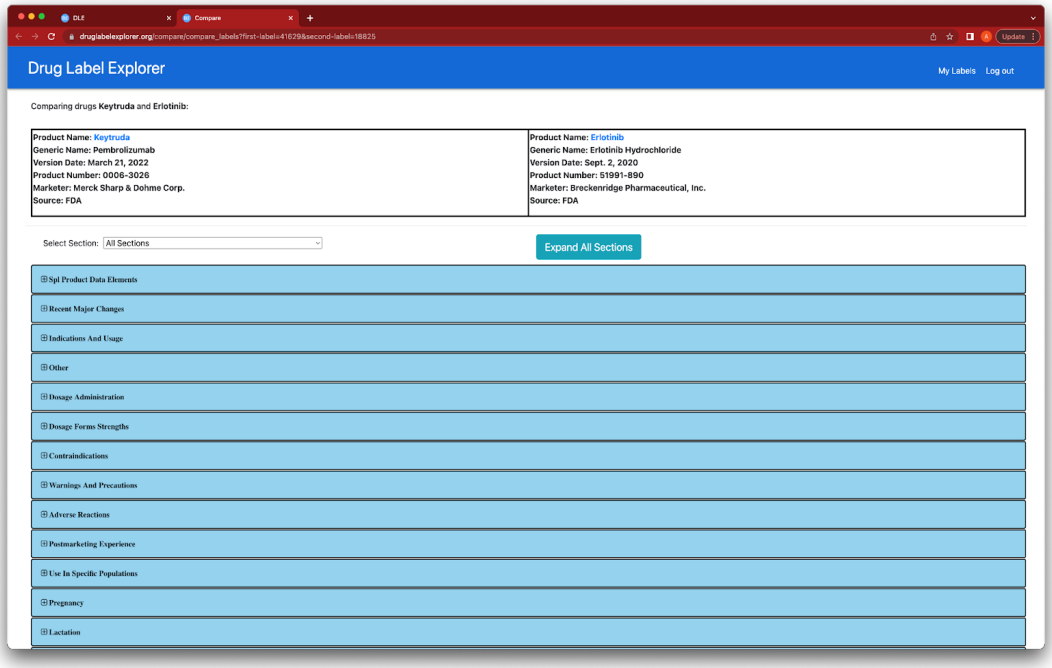


Figure 1.3: Drug Label Explorer Comparison View

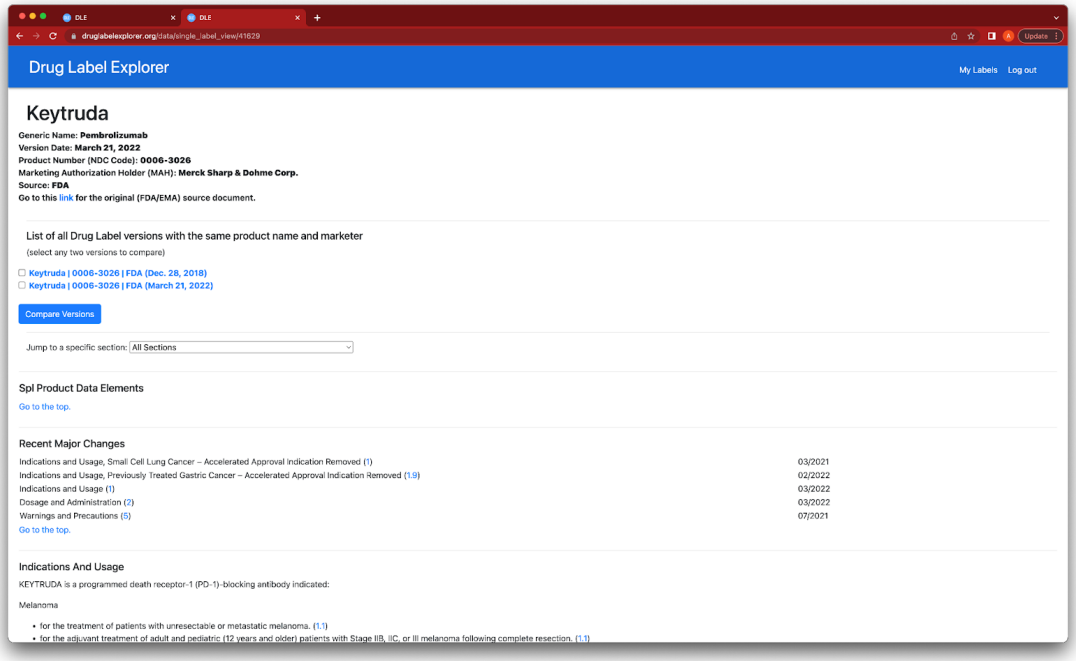


Figure 1.4: Drug Label Explorer Selected Drug Label View

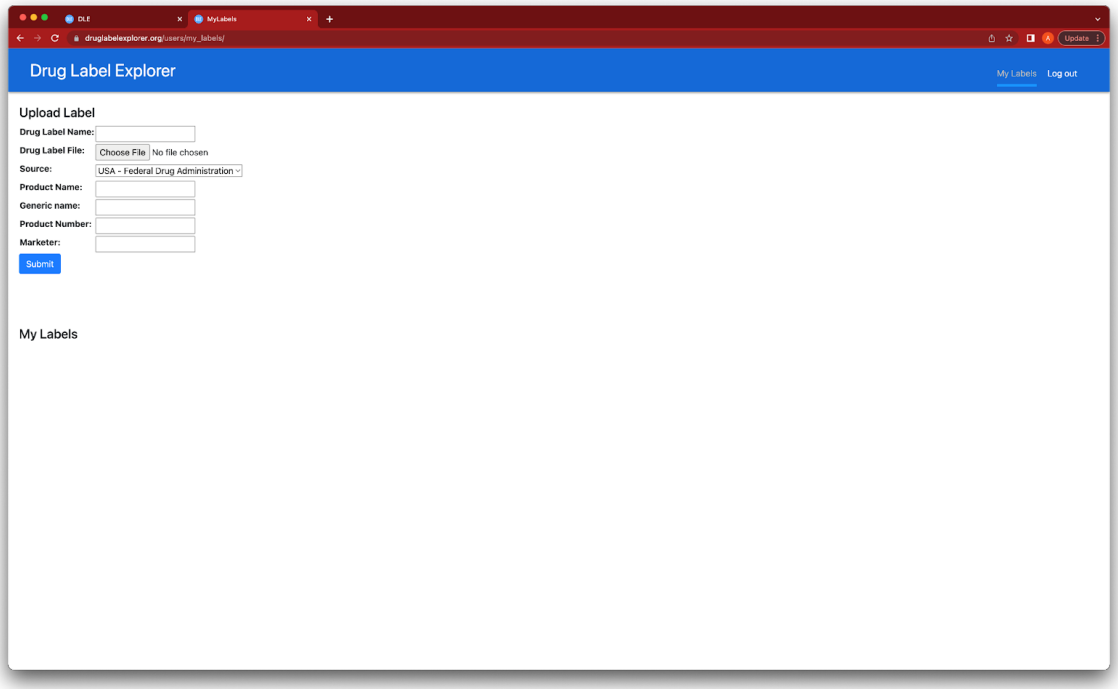


Figure 1.5: Drug Label Explorer Upload Custom Labels View

CONCLUSION

In this paper, we developed a software to extract transform and load (ETL) data dynamically from multiple resources such as FDA/DailyMed, EMA. Our software also provides the ability for registered users to upload their private drug labels. DLE then can filter through user input using search queries, and compare labels by section title and section contents. Some limitations in data parsing include mapping, as we stated 4% of FDA labels are put under an other labeled section. This works for a majority of use cases but in edge cases, we want to improve our ability to optimize identification of those sections.

In next steps, we plan to improve comparison and search features by implementing Bidirectional Encoder Representations from Transformers (BERT) model which is a deep learning model that can improve search performance and label comparisons by improving ability to understand free text more closely aligned to natural language and to further improve mapping of label content. Semantic search will be further improved using MedDRA ontology to account for biomedical terminology. DLE plans to expand to store drug label information for other agencies aside in different countries and in various languages in addition to English to further meet needs of DLE users on a global scale. Finally, we plan to implement visualizations to iteratively improve the user experience. Data visualization is a widely known concept and DLE would benefit from providing users with features that allow them to quickly identify patterns, make more insightful observations and recognize potential trends. The software package for this work in its most current state is available on the Drug Label Explorer website or on Github at https://github.com/DrugLabelExplorer/dle.

WORKS REFERENCED

[1] FDA: Code of Federal Regulations, Title 21, Vol.4, Chapter 1, Part 201-Labeling. Source:
https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?CFRPart=201
[2] European Commission: A Guideline on Summary of Product Characteristics (SmPC), September 2009.
Source:
https://ec.europa.eu/health/system/files/2016-11/smpc_guideline_rev2_en_0.pdf
[3] Lester, Jean, et al. (2013). Evaluation of FDA safety-related drug label changes in 2010.
Pharmacoepidemiology and Drug Safety, vol. 22.3, p302-305. Source:
https://onlinelibrary-wiley-com.ezp-prod1.hul.harvard.edu/doi/pdfdirect/10.1002/pds.3395
[4] Fang, Hong, et al. (2016). FDA drug labeling: rich resources to facilitate precision medicine, drug safety, and regulatory science. Drug Discovery Today, vol 21.10, p1566-1570. Source:
https://www-sciencedirect-com.ezp-prod1.hul.harvard.edu/science/article/pii/S1359644616
302240
[5] Moore, Thomas J., Sonal S., and Curt D. F. (2012). The FDA and new safety warnings.
Archives of Internal Medicine, vol. 172.1, p78-80.Source:
https://jamanetwork-com.ezp-prod1.hul.harvard.edu/journals/jamainternalmedicine/fullarticle/1108624
[6] Dusetzina, Stacie B., et al. (2012). Impact of FDA drug risk communications on health care utilization and health behaviors: a systematic review. Medical Care, vol. 50.6, p466.
Source:
https://oce-ovid-com.ezp-prod1.hul.harvard.edu/article/00005650-201206000-00002/HTML
[7] Seminerio, M. J., and M. J. Ratain. (2013). Are drug labels static or dynamic? Clinical
Pharmacology & Therapeutics, vol 94.3, p302-304. Source:
https://ascpt-onlinelibrary-wiley-com.ezp-prod1.hul.harvard.edu/doi/full/10.1038/clpt.2013
.109?sid=vendor%3Adatabase
[9] O. Nieminena, P. Kurkib, K. Nordstro. (2005). Differences in product information of

biopharmaceuticals in the EU and the USA: implications for product development. European Journal of Pharmaceutics and Biopharmaceutics, vol. 60.3, p319–32 Source: https://www-sciencedirect-com.ezp-prod1.hul.harvard.edu/science/article/pii/S0939641105 000780
[10] Rodriguez, T., et al. (2021). Medical Error Reduction and Prevention. National Center for
Biotechnology Information
Source: https://www.ncbi.nlm.nih.gov/books/NBK499956/
[11] Tariq, R., et al. (2021). Medication Dispensing Errors And Prevention. National Center for
Biotechnology Information
Source: https://www.ncbi.nlm.nih.gov/books/NBK519065/
[12] Delgado, N., etl al. (2019). Fast and accurate medication identification. npj Digital
Medicine, vol. 2.10
Source: https://www.nature.com/articles/s41746-019-0086-0#Sec6
[13] Jeetu, G., et al. (2010). Prescription Drug Labeling Medication Errors: A Big Deal for
Pharmacists. Journal of Young Pharmacists, vol 2.1, p107-111 Source: https://www.sciencedirect.com/science/article/abs/pii/S097514831021021X
[14] Davis, T. C., Federman, A. D., Bass, P. F., 3rd, Jackson, R. H., Middlebrooks, M., Parker, R. M., & Wolf, M. S. (2009). Improving Patient Understanding of Prescription Drug Label Instructions. Journal of General Internal Medicine, vol. 24.1, p57–62 Source: https://link.springer.com/article/10.1007/s11606-008-0833-4
[15] Shrank, W., Avorn, J., Rolon, C., & Shekelle, P. (2007). Effect of content and format of
prescription drug labels on readability, understanding, and medication use: a systematic
review. The Annals of pharmacotherapy, vol. 41.5, p783–801. Source: https://journals-sagepub-com.ezp-prod1.hul.harvard.edu/doi/pdf/10.1345/aph.1H582
[16] FDA Databases:
Source (Orange Book): https://www.accessdata.fda.gov/scripts/cder/ob/index.cfm and
Source (Drugs@FDA): https://www.accessdata.fda.gov/scripts/cder/daf/index.cfm
[17] NIH, DailyMed Database:
Source: https://dailymed.nlm.nih.gov/dailymed/index.cfm
[18] EMA, Medicines Database:
Source:
https://www.ema.europa.eu/en/medicines/what-we-publish-medicines-when-0

[19] FDA, Drug Safety-related Labeling Changes (SrLC) Database:
Source: https://www.accessdata.fda.gov/scripts/cder/safetylabelingchanges/
[20] RxList Website: https://www.rxlist.com (a WebMD owned product)
[21] ReedTech Website: https://www.reedtech.com
[22] WizMed Website: https://wizmed.com
[23] Cerner Website: https://www.cerner.com/solutions/drug-database (an Oracle owned
product)
[24] Drugs.com Website: https://www.drugs.com
[25] FDALabel: Full-Text Search of Drug Product Labeling: Source:
https://www.fda.gov/science-research/bioinformatics-tools/fdalabel-full-text-search-drug-pr
oduct-labeling#What%20is%20Included%20in%20Labeling
[26] Krist Shingjergji, Remzi Celebi, Jan Scholtes, Michel Dumontier Relation extraction from DailyMed structured product labels by optimally combining crowd, experts and machines, Journal of Biomedical informatics, Vol. 112, Oct 2021
Source: https://www.sciencedirect.com/science/article/pii/S1532046421002318
[27] Chen M.J., Vijay V., Shi Q., Liu Z.C., Fang H., and Tong W.D. "FDA-Approved Drug Labeling For the Study of Drug-Induced Liver Injury." Drug Discovery Today, vol. 16, p697-703 Source:
https://www.sciencedirect.com/science/article/abs/pii/S1359644611001668?via%3Dihub
[28] Shi, Y., Ren, P., et al. (2021). Information Extraction From FDA Drug Labeling to Enhance Product-Specific Guidance Assessment Using Natural Language Processing. Frontiers in Research Metrics and Analytics, vol 6. Source:
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8222600/pdf/frma-06-670006.pdf
[29] Hong, F., Harris, S., et al. (2016). FDA drug labeling: rich resources to facilitate precision medicine, drug safety, and regulatory science. Drug Discovery Today, vol 21. Source:
https://www.sciencedirect.com/science/article/pii/S1359644616302240?via%3Dihub
[30] Lindquist, L., Lindquist, L., et al. (2014) Unnecessary Complexity of Home Medication Regimens among Seniors. Patient Education and Counseling, vol 96.
Source: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4061206/
[31] Wu, L., Liu, Z. (2019). Study of serious adverse drug reactions using FDA-approved drug labeling and MedDRA. BMC Bioinformatics 20(S2). Source:
https://www.researchgate.net/publication/331748852_Study_of_serious_adverse_drug_reactions_using_FDA-approved_drug_labeling_and_MedDRA
[32] Fei, H., Ren, Y., Zhang, Y., Ji, D., Liang, X. (2021). Enriching contextualized language model from knowledge graph for biomedical information extraction. Briefing

in Bioinformatics, vol 22. Source:
https://academic.oup.com/bib/article-abstract/22/3/bbaa110/5854405
[33] MySQL. MySQL 8.0 Reference Manual. Source:
https://dev.mysql.com/doc/refman/8.0/en/
[34] Stoeva, M. (2021). EVOLUTION OF WEBSITE LAYOUT TECHNIQUES. Source:
https://www.researchgate.net/profile/Maya-Stoeva-2/publication/354675809_Evolution_of_Website_Layout_Techniques/links/61459d1c3c6cb3106977314d/Evolution-of-Website-Layout-Techniques.pdf

2. System Design

2.1 Tech Stack

The project essentially uses a LAMP (Linux, Apache, MariaDB, Python) technology stack which ultimately revolves around a Python application being deployed on AWS. The backend of the application is written in Python utilizing the Django framework with the front-end served via Django templates, effectively reducing most of the application logic within one single framework. The web server used for serving the web requests is Apache, which was ultimately chosen because one of the project members has many years of experience with the library, though many other alternative web servers can be a drop-in replacement. The Python application is served via the mod_wsgi Apache plugin. Lastly, the database the project team decided on is MariaDB. MariaDB was chosen due to the availability of its ColumnStore engine for fast analytics across large datasets as well as its familiar SQL syntax. Ultimately the team did not use the ColumnStore engine; after testing its suitability for the project it was decided to use the default InnoDB engine in MariaDB instead.

2.2 Tool Suite

The team is using Github as a primary tool suite for the project. Github Projects was chosen as the primary planning software. This ultimately leads us to use Github as the hosted git

Version Control System. Following this trend, the project uses Github Actions to orchestrate its CI pipeline. And Github Issues are used to track any bugs and action items that arise.

For testing, we are using the Django test harness which is an extension of Python's unittest module. The unit tests are executed using Github Actions on every pull request and on every merge into the main git branch. The results of the tests can be easily seen on Github with a green check mark indicating success and a red X indicating a test failure.

Utilizing all of Github's built-in tools reduces the amount of learning required with other existing 3rd party tools. Asynchronous communication is handled through Slack messages and email is used for coordinating meetings with Zoom conferencing for those who are not a part of the slack organization.

2.3 System Modules

The project requirements were broken up into modules to facilitate developers working on different parts of the application at the same time in a remote environment with team members in different time zones having different work schedules. The original technical requirements for this project, including estimates for when each feature were to be delivered are included in the Appendix. For the requirements, they are categorized into what roughly equates to Base Modules of the product.

- **Non-functional / DevOps**: We have a web application that is accessible on the internet that acts as a gateway to the features of the application. The web application supports encrypted traffic (https) and is able to be easily redeployed by an admin.

- **Data**: The website is backed by public data sources, cleaned, merged, and regularly updated. This data is served in a database that facilitates the queries created by the following features.

- **Users**: The website supports both anonymous and authenticated access, with certain features being restricted to users who have authenticated.

- **MyLabels**: When logged in, users can upload their Drug Labels and have them be queryable by the system. These uploaded Drug Labels are parsed and inserted into the Drug Label Explorer database. The user-uploaded Drug Labels are only accessible to the user who uploaded the Drug Label.

- **SearchForm**: One of the main features of the website is search form that allows Drug Labels to be queried. Capabilities for searching include: by drug brand name, by manufacturer, by label section, by agency and by generic drug name. This search form gives the user the ability to fine tune their results to get exactly what they need.

- **SearchResults**: The search results are displayed cleanly and the search terms are highlighted in the results. From this page the user is able to see blurbs from each search result with relevant keywords highlighted. From this page the user will also be able to click into a result to get more information, or select multiple results to compare them side by side.

- **SingleLabelView**: The user can switch into a detailed view of a single Drug Label. From this view the user will be able to see the entire content of the selected Drug Label.

- **CompareView**: The user can select 2 drug labels to view side by side, including separate drugs or different versions of the same drug. The view will clearly highlight areas that are similar and the differences.

2.4 Architectural Diagrams

Diagram showing the latest system architecture in AWS.

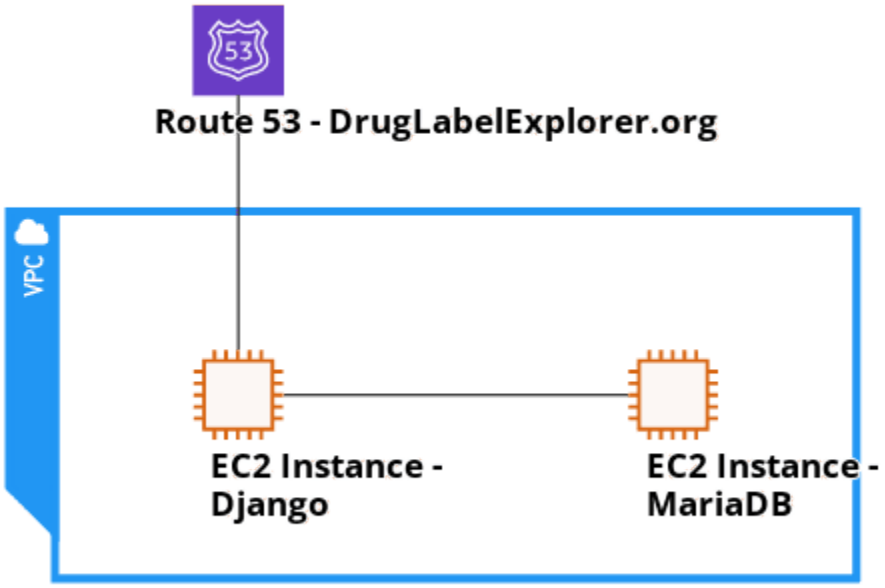
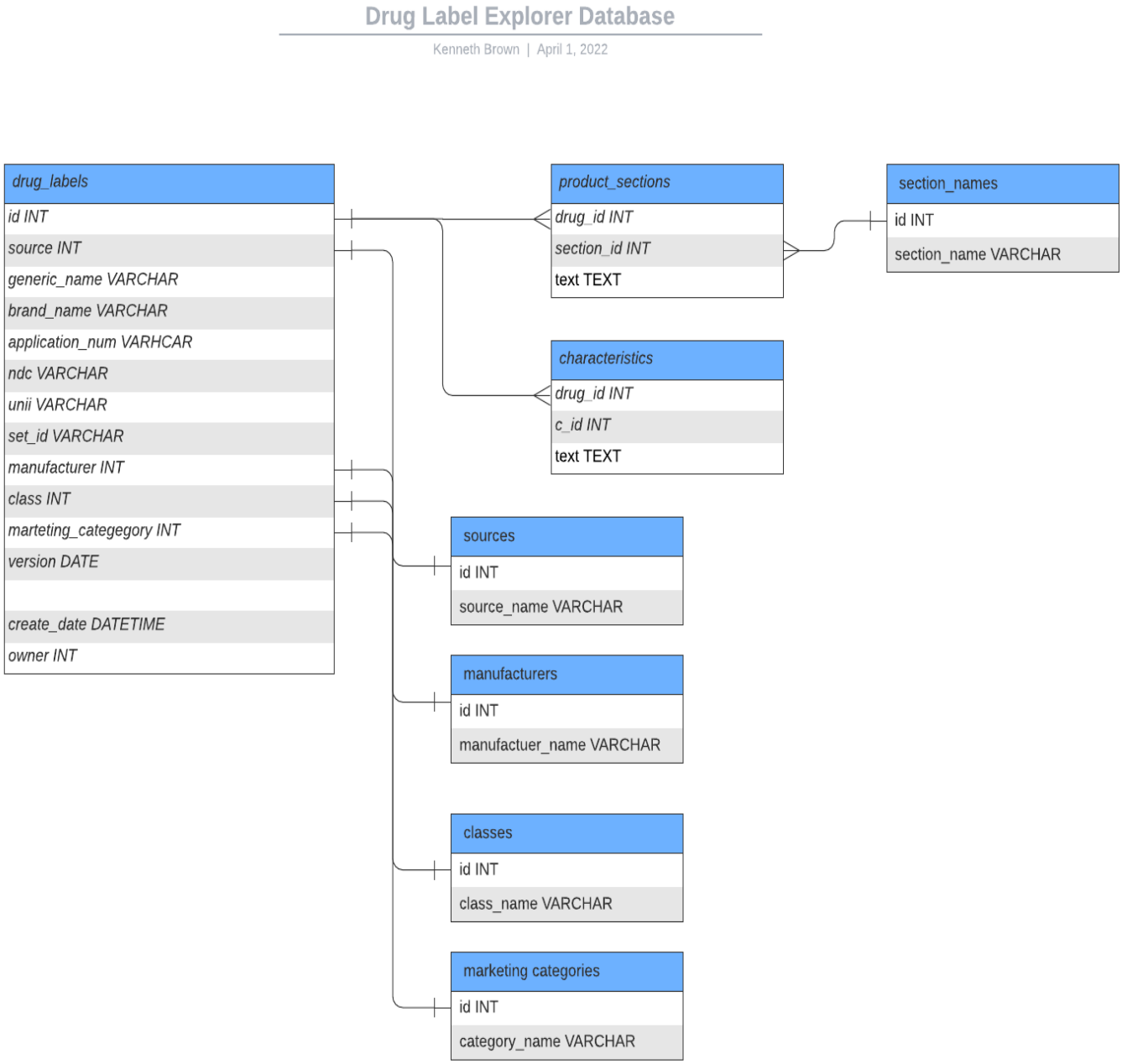


Diagram showing the latest class / database model for the DrugLabels.



3. Testing Results

For this project, our team conducted unit testing, performance testing, load testing, UI testing along with ad-hoc testing.

Unit testing is automatically performed on every pull request and on every code merge into the main git branch. Current unit test coverage is 63%. A report of the code coverage is included in the Appendix.

Performance testing was conducted periodically through development. The team was able to implement significant performance improvements, getting the average query time from 63 seconds per query to around 10 seconds per query. To facilitate performance testing a script was developed that can run the performance tests automatically with the command `python manage.py performance_tests` the test results are output to a CSV file and a png is created of the results. Some examples of the output are included in the Appendix.

There were some issues the team encountered with the database server in terms of handling a load of multiple queries at the same time. To ensure the reliability of the system, the team created a script that can perform load tests on the system. This is run with the command `python manage.py load_tests`. This helped the team ensure that the system can handle a reasonable load of queries concurrently.

Informal UI testing was performed with the Customer at the team meetings. The team implemented many of the suggestions that occurred at these back and forth sessions. In this way the team had an iterative approach to the User Interface implementation. The team provided the latest (uncompleted) iteration of each feature to the Customer as soon as it was ready. This allowed the Customer to give the team valuable feedback on the progress of the work, even before the features were completed.

Ad-hoc testing was also conducted as a part of the development process. Developers would – in addition to the unit testing – perform ad-hoc tests on the features under development. Developers would perform ad-hoc testing on the features assigned to them, but would also test the features assigned to the other developers during each sprint. The team had the goal to have at least one other developer help test the code before a feature was merged into the main repository branch.

4. Development Process and Lessons Learned

4.1 Meeting the Requirements

The team was able to complete most of the original specified requirements. The main requirements completed include parsing and loading of the data, being able to search the data, displaying the search results, being able to compare the drug labels and being able to upload drug labels that are only available to the individual. Some of the features that we were not able to implement due to time constraints include: being able to search via MedDRA synonyms, being able to save search queries, and being able to export the search results. The team worked with the customer to reprioritize the requirements on an ongoing basis through weekly meetings.

Holding weekly meetings with the customer where we displayed the current (work in progress) status of the project was invaluable in gathering feedback from the customer. This allowed us to focus our efforts on the areas of the project that provided the most value to the customer.

The team originally planned to meet weekly and have 1-week sprints. Partway through the project we added a meeting so we ended up having two meetings a week plus our customer meeting.

There were some changes to our original plans that our team had to deal with as the project progressed. One of the issues was the performance of the database. The team originally planned to use a MariaDB database with the ColumnStore engine as the database for the project. This turned out not to be the best choice after the implementation was not delivering the desired performance.

Another of the issues we encountered was loading of the data was more finicky than initially anticipated. We had in our model the concept of “sections” in the drug labels. But after working with the data and discussing with the customer, there was another level in the hierarchy, “subsections.” The team was able to make progress handling the subsections in the drug label data by working closely with the customer to refine the project requirements as the project progressed.

4.2 Estimates

Our initial time estimates were somewhat inaccurate. In pretty much all the cases, the amount of time spent working on a feature was about 2-3X that of the estimated time. This was mostly due to the time required to deal with “unforeseen” issues. So our initial estimates were somewhat of a “best case scenario” estimate. In the actual development process there were bugs that were introduced that added time to debug and fix. Also the amount of time iterating on a solution was not well captured by the estimates. For the features that we developed, they were improved over time so in some cases our estimates were “time estimates to deliver the first version” of a feature rather than “time estimates to deliver the final version” of a feature.

4.3 Risks

The Drug Label Explorer project enables users to quickly retrieve data on drug labels and their changes over time. This requires a significant amount of data aggregation across multiple data stores and regions to transform into an easily queryable dataset. Two identifiable risks are presented when dealing with data ingestion in this scenario, data mining across varied data stores that change with its region, and modeling the mined data in such a way that querying the data becomes a trivial task.

Effective data mining across multiple regions will enable the project to procure data and provide an effective strategy for data transformation. Originally, the project had requirements that the data be extracted from many different countries’ drug agencies. This presented a large risk because we would have to support parsing and translating an enumerable amount of languages and label formats. The project team was able to de-risk this requirement significantly by scoping the project to 2 drug agencies, FDA and EMA. With the data sources limited to just two agencies, the project team can more effectively mine data from the respective agency’s datastore without having to program the mining tools to take into account different languages and additional label formats.

Once data is extracted from the drug agencies datastore, it will need to enter a transformation process that will parse, clean, and store the data in a queryable format. The risk that comes with ETL on mined data from these datastores is that the parsing step will have to make tradeoffs between acquiring more data points or maintaining high accuracy within the extracted data. This risk is apparent when we compare the FDA XML files against the EMA PDF files. Both of these agencies provide data that are loosely structured and when parsing such data the aforementioned risk is apparent. The project team was able to reduce the risk and complexity of this ETL step by reducing the scope of the features on the dataset

when persisting the data. With the emphasis on high-accuracy and high-quality features on the dataset, the parsing and storing of the data is now more focused and simpler to execute.

In addition to the aforementioned risks, there are also the inherent risks with the limitation of web scraping tools available for working with PDFs. This risk is highlighted by the use of PDF templates that the implementation requires when expecting a PDF of a certain format to be parsed. Inevitably, there will be PDFs that don't align perfectly with the template and will reduce the accuracy of the parsed data. The project team has mitigated this risk by assuming that the accuracy levels of parsed data may not be 100%, and any errors will be logged and categorized internally to further improve the system. The team plans to review the actual results with the customer every week to achieve customer satisfaction within the timeframe of the project.

While we anticipated the risks in working with the PDF data from the EMA datasource, in reality parsing the XML data from the FDA datasource turned out to be more challenging. It turned out that the FDA XML data is structured in a less standardized way than the EMA PDF data, which made parsing the FDA data more complex.

One of the main risks that was unanticipated was the performance of the database. If we were not able to query the data in a reasonably timely manner, the whole project would be a bust. The team did a good amount of planning initially to come up with what we thought was a good solution for how to store and query the data, but testing showed that we needed to change course. As a result, we were able to deliver a project that has a reasonable query time. Unfortunately, this took more development cycles than anticipated.

Another unanticipated risk is if a team member gets sick, takes a vacation or is unable to work on the project for unforeseen personal reasons. This was not a major problem for our team. But there might have been one or more occasions where a developer's absence slowed down the project's progress a little bit.

4.4 Team Dynamic

The Drug Label Explorer Team has a shared goal of developing the best possible outcome to improve the solution for our client. To do so we agreed on utilizing open communication which will be primarily conducted through Slack to account for differing time zones and personal work patterns, and we utilize Google Drive and GitHub for deliverables, we have set

up a shared Google Drive and a GitHub Project to this end. Client meetings are recorded and shared throughout the project. All team members are encouraged to discuss issues and problems that may arise. DLE plans to take advantage of members' unique abilities to maintain efficiency while maintaining a collaborative approach throughout all sections of the project.

In terms of conflict resolution, DLE collectively prefers a minimal contact strategy meaning issues will be raised between conflicting parties and only be brought to the entire group and/or the teaching staff if conflicts cannot be resolved or compromised internally. This strategy aligns with our shared value of open communication. Group decisions made by DLE will have a hybrid of majority rule and guided by members with advanced expertise in a given topic (consensus decision-making).

DLE originally planned to meet twice per week: on Saturday (internally discuss varying relevant agendas) and on Friday (with the Customer to discuss relevant issues/agenda). We ended up adding a meeting on Tuesday to facilitate the project's progress.

The team initially planned to have 9 weekly Sprints, running Thursday to Thursday, showing the latest progress with the Customer on Friday. The goal is to have open communication with the Customer. Each week we sought and gained valuable feedback and discussed any issues as they arose to help facilitate the success of the project.

Having the weekly Sprints did not really help us as a team. Instead we integrated features into the main branch in an ad-hoc manner whenever they were ready.

The team initially decided that each team member would "self manage" meaning that we would create our own Tickets and update our progress on the Github Project Workboard. This did not work out as well as originally hoped as some team members would forget to perform these management tasks. Not having a designated "project manager" might have also hurt the team a little in that there was no one to "keep team members accountable" if they were following through with their tasks in the expected timeframe. Overall this wasn't really an issue for our team. But if this was a larger project or if there were concrete deliverables that were more time sensitive, this might have been a larger issue.

5. Appendix

5.1 Technical Requirements

For these requirements, Week 1 Starts on March 3rd, 2022, and the requirements are expected to be delivered on or before Milestone 2 (April 7th) or Milestone 3 (May 5th) as listed.

Deliverable 1:
There is a website is available on the public internet that allows people to run queries on drug labels
Category: Non-functional
Estimate (hours): 8
Start week: 1
Milestone delivery: 2

Deliverable 2:
Website is protected by industry standard TLS encryption
Category: Non-functional
Estimate (hours): 1
Start week: 1
Milestone delivery: 2

Deliverable 3:
Website can handle a small number of concurrent users - tens
Category: Non-functional
Estimate (hours): 0, because a small number of Users are handled automatically, and if we wanted to handle millions of users, that would take more time.
Start week: 1
Milestone delivery: 2

Deliverable 4:
Instructions provided to deploy / redeploy all System components on Amazon Web. AWS CloudFormation template(s) to assist with the deployment of System components Services (AWS)
Category: Non-functional
Estimate (hours): 5

Start week: 1
Milestone delivery: 3

Deliverable 5:
Database with security measures to protect the data stored in the database including encryption at rest and encryption in transit (via SSL)
Category: Non-functional
Estimate (hours): 3
Start week: 1
Milestone delivery: 2

Deliverable 6:
Database instance is configured to automatically run Snapshot backups daily, keeping a 30 day rolling window of backups
Category: Non-functional
Estimate (hours): 1
Start week: 6
Milestone delivery: 3

Deliverable 7:
Response times of the system are within reason
Category: Non-functional
Estimate (hours): 12
Start week: 6
Milestone delivery: 3

Deliverable 8:
System has access to FDA Drug Label data from DailyMed (SPL/XML), to include the latest versions for all approved prescription drug labels
Category: Data
Estimate (hours): 40
Start week: 1
Milestone delivery: 2

Deliverable 9:
System has access to FDA Drug Label data from DailyMed (SPL/XML), to include ALL historical versions for all prescription drug labels for the previous 3, 5, or 7 years (TBD)
Category: Data (History)
Estimate (hours): 20
Start week: 6
Milestone delivery: 3

Deliverable 10:
System automatically refreshes data from its data sources at specified cadence (daily, weekly, monthly)
Category: Data (Refresh)
Estimate (hours): 20
Start week: 6
Milestone delivery: 3

Deliverable 11:
System accesses data from EU data source (PDFs), to include the latest version for all prescription drug labels
Category: Data
Estimate (hours): 75
Start week: 3
Milestone delivery: 2

Deliverable 12:
Drug label data can be accessed using MedDRA terms
Category: Data
Estimate (hours): 35
Start week: 2
Milestone delivery: 3

Deliverable 13:
Data from all data sources is standardized using the Findable, Accessible, Interoperable, Reusable (FAIR) principles. This will be done with a uniform schema and search tools designed to directly interface with such.
Category: Data

Estimate (hours): 12
Start week: 2
Milestone delivery: 2

Deliverable 14:
Drug label search functionality is available to an unregistered / guest / null User
Category: Users
Estimate (hours): 0
Start week: 1
Milestone delivery: 2

Deliverable 15:
Basic user authentication including sign-up, sign-in, and password reset using email allows for additional features such as uploading labels, saving queries, etc.
Category: Users
Estimate (hours): 20
Start week: 1
Milestone delivery: 2

Deliverable 16:
Ability to upload labels conforming to a supported type: FDA/XML, EU/PDF; labels are only available to the single user (by default)
Category: MyLabels
Estimate (hours): 18
Start week: 4
Milestone delivery: 3

Deliverable 17:
Ability to share saved labels with other registered users; after selecting a label and choosing an email address, the system will send an email with a link to a page in the system that shows the label
Category: MyLabels
Estimate (hours): 5
Start week: 6
Milestone delivery: 3

Deliverable 18:
Sharing user-uploaded drug label, grants access to the registered user with the recipients email address
Category: MyLabels
Estimate (hours): 3
Start week: 6
Milestone delivery: 3

Deliverable 19:
User-uploaded drug labels show up in the user's search results along with other drug labels; only the user who uploaded the label or other users with whom the label was shared have access
Category: MyLabels
Estimate (hours): 4
Start week: 6
Milestone delivery: 3

Deliverable 20:
Ability to Save searches
Category: MyQueries
Estimate (hours): 6
Start week: 7
Milestone delivery: 3

Deliverable 21:
Main page of the application has a SearchForm area that includes the functionality for searching the Drug Labels. In general this can include drop-downs, checkboxes, text fields, etc.
Category: SearchForm
Estimate (hours): 35
Start week: 1
Milestone delivery: 2

Deliverable 22:
Ability to limit searches to FDA Drug Labels, EU Drug Labels or both
Category: SearchForm

Estimate (hours): 2
Start week: 1
Milestone delivery: 2

Deliverable 23:
Ability to Search by Product (Generic and/or Brand Name)
Category: SearchForm
Estimate (hours): 3
Start week: 1
Milestone delivery: 2

Deliverable 24:
Ability to Search by Application number, DEA schedule, NDC, UNI code, SET ID
Category: SearchForm
Estimate (hours): 3
Start week: 4
Milestone delivery: 2

Deliverable 25:
Ability to Search by Product Characteristics (color, imprint, shape, size, scoring, etc)
Category: SearchForm
Estimate (hours): 3
Start week: 4
Milestone delivery: 2

Deliverable 26:
Ability to Search by drug Marketer
Category: SearchForm
Estimate (hours): 3
Start week: 4
Milestone delivery: 2

Deliverable 27:
Ability to Search by Label Section
Category: SearchForm
Estimate (hours): 3
Start week: 4

Milestone delivery: 2

Deliverable 28:
Ability to Search by MedDRA terms
Category: SearchForm
Estimate (hours): 3
Start week: 4
Milestone delivery: 2

Deliverable 29:
Ability to perform wildcard search on drug label data when searching within drug label categories
Category: SearchForm
Estimate (hours): 4
Start week: 6
Milestone delivery: 3

Deliverable 30:
Ability to perform proximity search — a user should be allowed to search for drug label terms that are within a specified distance from each other (e.g. number of words apart, within the same paragraph, or within the same section)
Category: SearchForm
Estimate (hours): 15
Start week: 7
Milestone delivery: 3

Deliverable 31:
Ability to Filter Search Results - Pharmacologic class
Category: SearchForm
Estimate (hours): 1
Start week: 5
Milestone delivery: 2

Deliverable 32:
Ability to Filter Search Results - marketing categories
Category: SearchForm

Estimate (hours): 1
Start week: 5
Milestone delivery: 2

Deliverable 33:
User has some ability to adjust what data columns are displayed from the query results
Category: SearchForm-Results
Estimate (hours): 8
Start week: 3
Milestone delivery: 2

Deliverable 34:
Ability to Group Search Results by Generic Name
Category: SearchForm-Results
Estimate (hours): 2
Start week: 3
Milestone delivery: 2

Deliverable 35:
Ability to Group Search Results by Manufacturer
Category: SearchForm-Results
Estimate (hours): 1
Start week: 3
Milestone delivery: 2

Deliverable 36:
Ability to Group Search Results by Country
Category: SearchForm-Results
Estimate (hours): 1
Start week: 3
Milestone delivery: 2

Deliverable 37:
Ability to Group Search Results by Marketing Category (i.e. Application Type)
Category: SearchForm-Results
Estimate (hours): 1
Start week: 3

Milestone delivery: 2

Deliverable 38:
Ability to specify “latest version” or “all versions” for the drug labels in the search results. Drug label versions are derived from the date the document was last updated.
Category: SearchForm
Estimate (hours): 2
Start week: 2
Milestone delivery: 2

Deliverable 39:
Ability to have multiple search criteria. Ability to apply up to 5 search criteria with AND operators.
Category: SearchForm
Estimate (hours): 5
Start week: 2
Milestone delivery: 2

Deliverable 40:
After the search is executed, the search results are displayed to the user. The search results view should display a list of the matching drug labels. The search results may be paginated when they exceed a specified number of drug labels.
Category: SearchResults
Estimate (hours): 50
Start week: 2
Milestone delivery: 2

Deliverable 41:
The Search query parameters used in the search are highlighted in the SearchResults when present
Category: SearchResults
Estimate (hours): 3
Start week: 6
Milestone delivery: 3

Deliverable 42:

A details page for the drug label is shown after the user clicks on an item from the search results.
Category: SingleLabelView
Estimate (hours): 20
Start week: 4
Milestone delivery: 3

Deliverable 43:
The Search query parameters used in the search are highlighted in the SingleLabelView
Category: SingleLabelView
Estimate (hours): 3
Start week: 6
Milestone delivery: 3

Deliverable 44:
In the SearchResults there is the ability to select two labels. After selecting two labels, the user can then compare the labels.
Category: SearchResults - Compare
Estimate (hours): 4
Start week: 6
Milestone delivery: 3

Deliverable 45:
Side-by-Side Comparison with "Track Changes" View (Two labels). As a user scrolls through the page, both sides of the view should be in sync.
Category: Compare
Estimate (hours): 38
Start week: 3
Milestone delivery: 3

Deliverable 46:
Search results are automatically highlighted in the side by side comparison view of the drug labels
Category: Compare
Estimate (hours): 2
Start week: 5
Milestone delivery: 3

Deliverable 47:
Ability to navigate to the VersionHistoryView from the SearchResults view
Category: SearchResults - VersionHistory
Estimate (hours): 2
Start week: 4
Milestone delivery: 3

Deliverable 48:
Ability to navigate to the VersionHistoryView from the SingleLabelView
Category: SingleLabelView - VersionHistory
Estimate (hours): 2
Start week: 4
Milestone delivery: 3

Deliverable 49:
A Version History View page is displayed showing changes to a drug label over time
Category: VersionHistory
Estimate (hours): 32
Start week: 3
Milestone delivery: 3

Deliverable 50:
Search results automatically highlighted in the version history page
Category: VersionHistory
Estimate (hours): 2
Start week: 6
Milestone delivery: 3

Deliverable 51:
Ability to export selected columns from multiple labels from the SearchResults
Category: Export
Estimate (hours): 3
Start week: 8
Milestone delivery: 3

Deliverable 52:
Ability to export Label Comparison to HTML
Category: Export
Estimate (hours): 2
Start week: 8
Milestone delivery: 3

Deliverable 53:
Ability to export the Version History View to HTML
Category: Export
Estimate (hours): 2
Start week: 8
Milestone delivery: 3

Deliverable 54:
All export HTML pages include highlighting of the search parameters, when present
Category: Export
Estimate (hours): 1
Start week: 8
Milestone delivery: 3

5.2 Wireframes

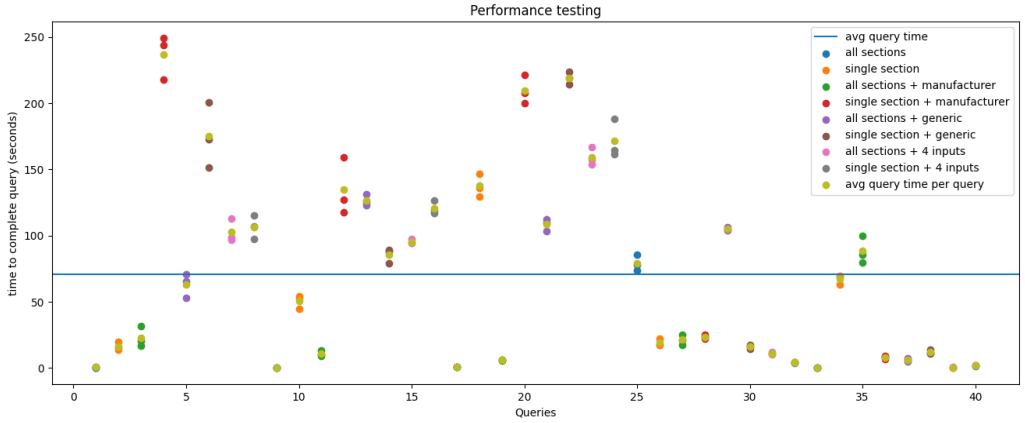
Wireframes from our client, David Edelen

5.3 Unit Test Code Coverage Report

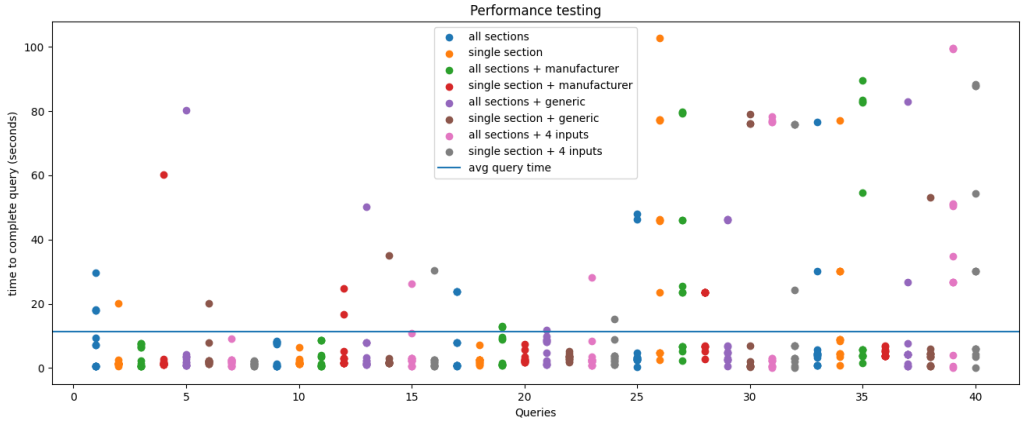
Name	Stmts	Miss	Cover
compare/apps.py	4	0	100%
compare/models.py	2	0	100%
compare/tests.py	1	0	100%
compare/urls.py	4	0	100%
compare/util.py	87	79	9%
compare/views.py	130	120	8%
data/apps.py	4	0	100%
data/constants.py	2	0	100%
data/management/commands/load_ema_data.py	185	44	76%
data/management/commands/load_fda_data.py	235	79	66%
data/management/commands/update_latest_drug_labels.py	27	2	93%
data/models.py	21	0	100%
data/tests.py	62	0	100%
data/urls.py	4	0	100%
data/views.py	22	16	27%
dle/settings.py	29	1	97%
dle/urls.py	7	1	86%
manage.py	12	2	83%
search/apps.py	4	0	100%
search/models.py	18	2	89%
search/search_constants.py	2	0	100%
search/services.py	84	36	57%
search/tests.py	23	0	100%
search/urls.py	3	0	100%
search/views.py	25	16	36%
users/apps.py	4	0	100%
users/forms.py	10	0	100%
users/models.py	13	1	92%
users/tests.py	58	2	97%
users/urls.py	4	0	100%
users/views.py	80	35	56%
TOTAL	1190	436	63%

5.4 Performance Test Results

Initial performance testing indicated an average query time of 63 seconds per query.



Updated hardware in conjunction with query improvements led to an average query time of around 10 seconds per query.



After additional improvements the team was able to get this down to 2.7 seconds average query time per our performance test benchmark tool.