# Bay Area Municipal Code Repository

By Kamron Afshar & Michael Schulze

# Goals

The goals of this project were to set up a system that would update a database that contained all municipal codes in the bay area and identify changes to those codes over time.

These municipal codes were being stored across 4 multi-municipal websites and 4-5 other small or independent websites. Since there was no API offered for any of the websites our only option was to scrape the data out of these sites then collect that data as S3 documents and in a redshift database.

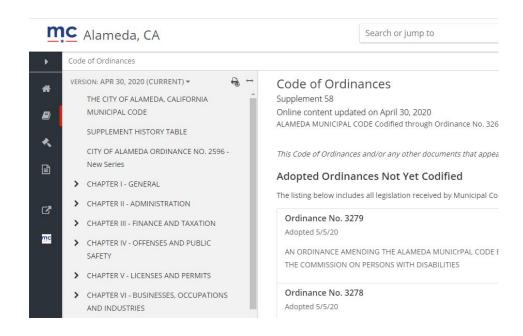
# Step 1: Locate the municipal codes

We found this website <a href="https://igs.berkeley.edu/library/california-local-government-documents/codes-and-charters">https://igs.berkeley.edu/library/california-local-government-documents/codes-and-charters</a> that located almost all of the municipal codes in their respective websites so we scraped this information and put it in a table so we could identify which municipalities kept their code in which website.

There were four main websites that hosted the majority of the municipalities codes: municode.com, codepublishing.com, qcode.us, and amlegal.com. Each website used a different format for displaying the codes so we had to build four different scrapers, one for each website.

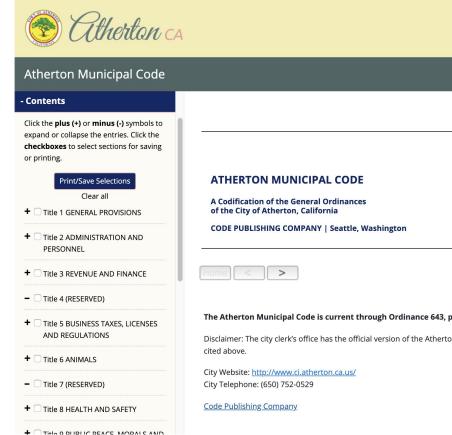
# Step 2a: Scraping municode.com

- Use Selenium to click through each document and extract text.
- Write level 2 documents to S3
- combining level 3 and lower docs into level 2.
- Only crawls page if listed update date has changed



# Step 2b: Scraping codepublishing.com

- Check all the boxes then save the document to our base location.
- Parse document into the separate titles then save those titles (level 2) each as a document in S3.



# Step 2c: Scraping qcode.us

- Have to travel exhaustively through tree-like structure to get text data.
- Combine all that text data together as I crawl and add each title (level 2) as a separate document to S3

### **Burlingame Municipal Code** Burlingame, California



Current through Ordinance 1976 and the May 2020 code supplement. For more recent amendments to this code, please contact the city.

purposes only. Please read the full disclaimer.

- Search
- · Statutory References

### Links:

· Burlingame, California - Home Page

### Contact:

City Clerk: (650) 558-7203

# **Burlingame Municipal Code**



This document is provided for informational

Municipal Code Table of Contents

Ordinance List

Title 16 (RESERVED) Title 17 FIRE

Title 14 (RESERVED

Title 7 (RESERVED)

**Title 9 ANIMALS** 

**Title 18 BUILDING CONSTRUCTION** Title 19 (RESERVED)

**Title 1 GENERAL PROVISIONS** Title 2 ADMINISTRATION

**Title 4 REVENUE AND FINANCE** Title 5 (RESERVED)

**Title 8 HEALTH AND SANITATION** 

Title 13 VEHICLES AND TRAFFIC

**Title 15 WATER AND SEWERS** 

Title 3 OFFICERS, BOARDS, COMMISSIONS AND PERSONNEL

**Title 6 BUSINESS LICENSES AND REGULATIONS** 

**Title 10 PUBLIC PEACE, MORALS AND SAFETY** Title 11 TREES AND VEGETATION **Title 12 STREETS AND SIDEWALKS** 

Title 20 (RESERVED)

**Title 21 HISTORIC RESOURCE PRESERVATION** 

**Title 22 SIGNS** 

**Title 23 SWIMMING POOLS, HOT TUBS AND SPAS** 

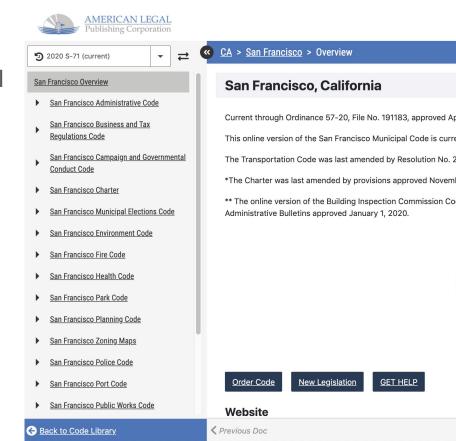
Title 24 RESERVED Title 25 ZONING

**Title 26 SUBDIVISIONS** 

View the mobile version.

# Step 2d: Scraping amlegal.com

- Have to expand and collapse the different "codes" (level 2) then select all parts of that level two so I can save the document to our base location.
- Upload each level 2 as its own document to S3.



# Step 3: S3 document

- Level 2 documents are written to S3 when crawled or pieced together
- Documents are identified by an S3 key or filename
- Once written, S3 keys are included in the redshift table to allow access

# Step 4: Redshift

## Columns:

- Name of municipality
- Date the code was updated
- Title of the level 2 section
- Did this document contain information on zoning
- The calculated "diff" score
- The S3 key that links to the level 2 document

# Step 5: Repeat

We need to run this executer script in order to update the data and the interval that this occurs at can depend on how frequently one expects the data to change. Our recommendation is to run this script monthly.

# Step 6: In the future...

- Use NLP text parsing to further subdivide our level 2 documents in S3 so we can narrow our text search.
- Reverse engineer scrapers so they travel to a designated subsection that the user wants to look at in the code.
- Create a more sophisticated text difference metric to better identify how much the document is actually changing.
- Do NLP analytics on the documents to discover interesting patterns in zoning policy or legal phrasing. How do different cities word and treat housing concerns
- Parallelize scrapers