TEXAS LAW DATABASE

Team: For-git about it

Members: Jimmy Kettler, Eliya Shabanov, Ian Hays, Keyon Mohebzad, Paul Strong, Michael Dong

Contents

Introduction

Problem

The democratic process works best if there is a well-informed citizenry. Given more information, voters can make more informed decisions on their support for political candidates. However, there currently does not exist a well-made central database containing information concerning the Texas Senate. There is no place to see comprehensive information on bills, senators, committees and the relationship between them. Without such a database, it is difficult for citizens to hold their government accountable. The ability for voters to see a candidate's voting record allows them to decide for themselves whether a candidate is suitable for the job. While such databases exist for the federal government (https://www.govtrack.us/ is an example), the only place where one can access a database for bills going through the Texas legislature is through the official Texas legislature website, which is lacking in information. The Texas Law Database exists to provide such a service, to open source democracy in Texas, and to serve as a source of information so that voters can learn about bills, legislators, and committees in order to stay informed about the democratic process in Texas.

Use Cases

The Texas Law Database contains the following things:

Information about specific bills that are going through, or have gone through, the Texas Senate, including a brief summary, author, status, a link to the full text of the bill, the committee that the bill went through, and a breakdown of the votes for the bill.

Information about individual senators, including a brief biography, their voting history, a list of committees they chair or are a member of, and a link to their facebook and/or twitter, if they have one.

Information about Senate committees, including a description, their members, and a list of all bills to have gone through that committee.

Users of the Texas Law Database can learn about the most recent bills to go through committee or a floor vote, or they can look through the lists of senators and committees and see information about the bills they have voted on.

Users can use the Texas Law Database to see Senator's voting records over time, and thus get a better understanding of a senator's position on issues. Voters can then use this information to affect their voting decision.

Users can see a list of bills that passed or failed and gain a greater understanding of the legal trends for the Senate as a whole. Users in the political field can use this information to make informed predictions on proposed bills or bills to be proposed.

The net effect of this information is to allow voters in Texas to contribute to and to learn about the Texas senate, to inject some much needed openness in the democratic process, and to give voters and citizens in Texas the tools to make informed decisions about their Senate.

Backend Design

Design Overview

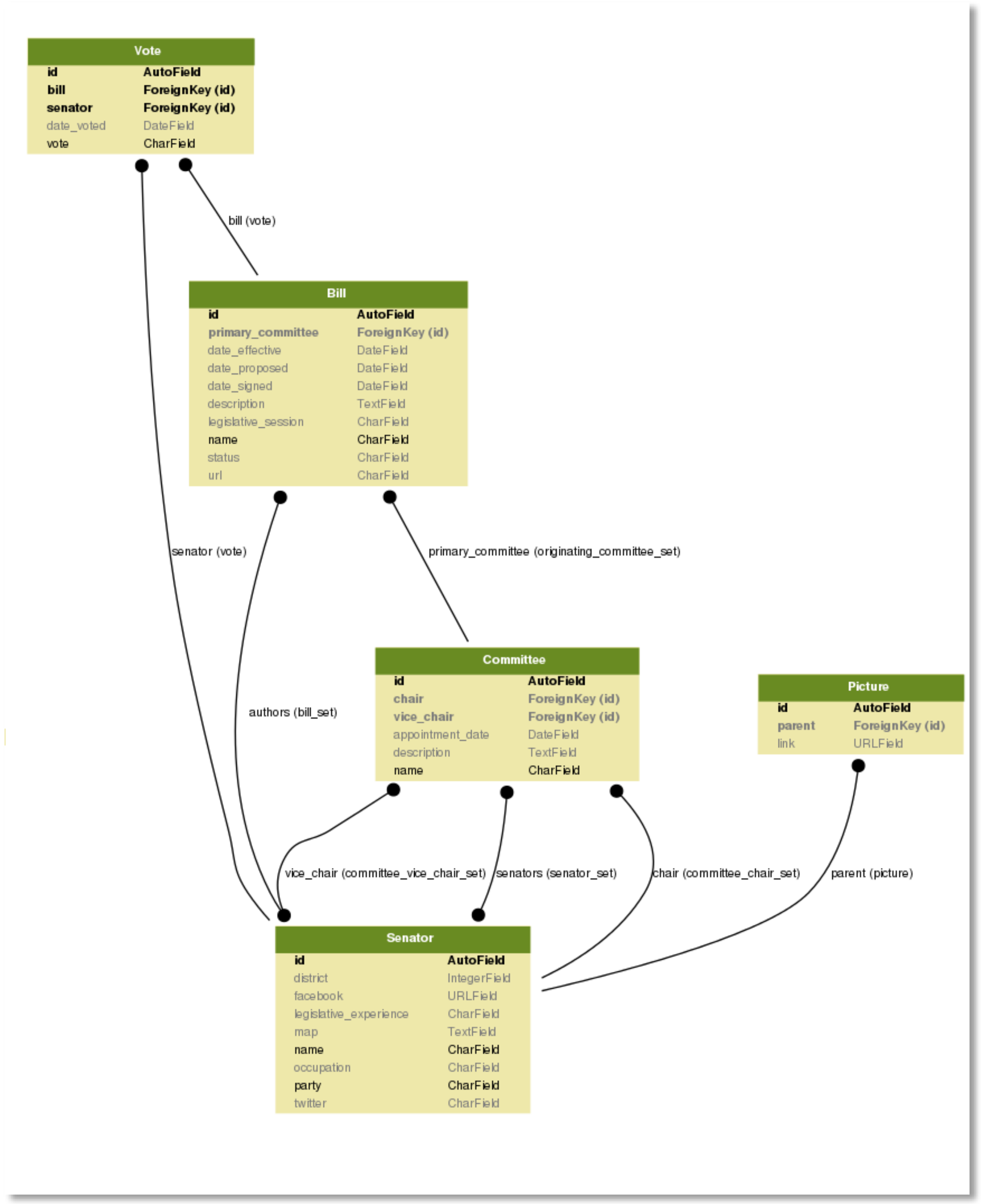
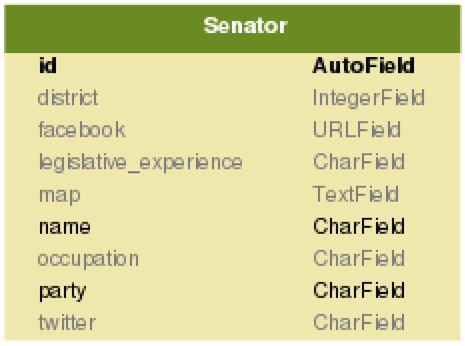


FIGURE 1: UML OVERVIEW OF MODELS

Django Models

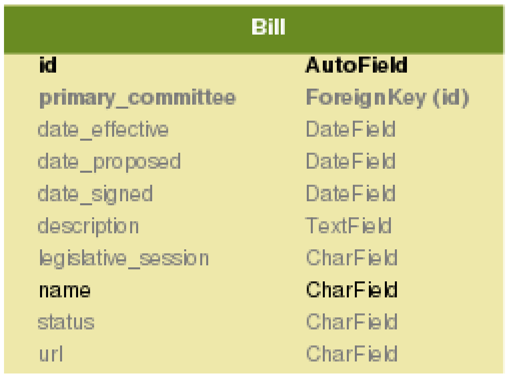
Senator Model



Senator	
id	**AutoField**
district	IntegerField
facebook	URLField
legislative_experience	CharField
map	TextField
name	CharField
occupation	CharField
party	CharField
twitter	CharField

The senator model models each of the 31 individual senators in the Texas Senate. These are the legislators in the Texas Senate who author, sponsor, and vote on bills, whether in committees or on the floor. The primary key is an auto-generated integer ID supplied by Django. Attributes include their name, district, party, occupation, legislative experience, a map of where their office is located, and their facebook and twitter pages if they have one.

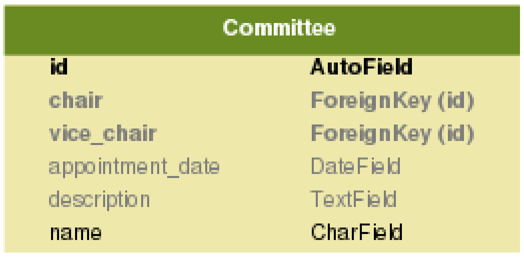
Bill Model



Bill	
id	**AutoField**
primary_committee	**ForeignKey (id)**
date_effective	DateField
date_proposed	DateField
date_signed	DateField
description	TextField
legislative_session	CharField
name	CharField
status	CharField
url	CharField

The bill model models specific bills submitted to committees or the senate floor at large. These bills are bills that have gone to a vote in committees or on the floor. As with senators, their primary key is an auto-generated integer supplied by Django. Attributes include the name of the bill, the legislative session in which the bill was proposed, the date it was proposed, and if passed, the date it was signed into law and the date it was effective. Attributes also include its status, a link to its full text, and a brief description on the contents of the bill.

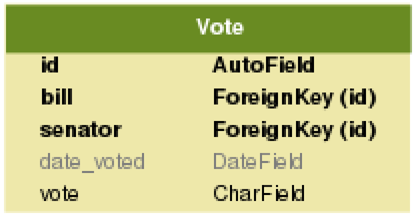
Committee Model



Committee	
id	**AutoField**
chair	**ForeignKey (id)**
vice_chair	**ForeignKey (id)**
appointment_date	DateField
description	TextField
name	CharField

The committee model models the 31 standing committees in the Texas Senate. Every bill must go through committee, where it is debated, amended, and voted on before it goes to the floor where it is voted on by the Senate as a whole. As with senators and bills, the primary key is an auto-generated integer supplied by Django. Attributes include its name, a description, and the date that the committee was first formed.

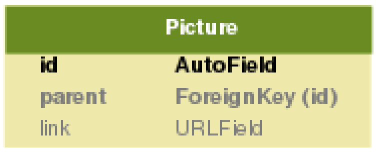
Vote Model



Vote	
id	**AutoField**
bill	**ForeignKey (id)**
senator	**ForeignKey (id)**
date_voted	DateField
vote	CharField

The vote model models the votes that senators cast for bills. Senators may vote Aye or Nay, or they may be present but not voting, or absent from the vote. We store the actual value of the vote itself as a tuple of strings, a 3 letter abbreviation for the value of the vote, and its full English value. We also store as an attribute the date of the vote.

Picture Model



Picture	
id	**AutoField**
parent	ForeignKey (id)
link	URLField

The picture model is there to represent the one to many relationship between senators and pictures. Instead of having one picture link per senator, the picture model allows us to display a slideshow of pictures for each senator.

Senator and Bill Relationship

Bills are authored by Senators, and although there are co-authors, we have chosen to only include the main author, for simplicity's sake. A bill can have a long list of co-authors, some of which may not be senators, and thus, as an initial step, we have decided only to include main authors. Senators can author multiple bills. Therefore, we have a many to one relationship between bills and senators.

Senator and Committee Relationship

There are 31 standing committees in the Texas Senate. Each committee is chaired by a senator and vice chaired by a senator, though those positions may be vacant at any given time. Each committee also has some number of senators as members, usually less than 10. A senator may be part of any number of committees. Therefore, there exists a many to many relationship between senators and committees. Since there are multiple types of relationship between senators and committees: members, vice chairs, and chairs, a senator's membership of a committee is modeled by the three different sets a senator to committee relation can belong to: committee_chair_set, committee_vice_chair_set, and committee_senators_set.

Bill and Committee Relationship

Each bill must go through a committee. While a bill may go through multiple committees before going to the floor for a vote, and indeed, every bill must go through the calendar committee to be placed on the floor for a vote, each bill mainly falls under the purview of one committee. This is the committee which we have chosen to model. Each committee oversees many bills. Therefore, there exists a many to one relationship between bills and committees.

Django Unit Tests

Django unit tests, found at cs373-idb/idb/tests, aims to comprehensively test the capabilities of the Django model. For each of the models, we first set up a test case for the model by creating, using Django methods, a test member with the desired attributes and relationships. This simulates the way Django itself uses these methods to populate the database. We then use Django's methods to retrieve a member, and we check that the values in Django's database match what is expected. In this way, we are able to test Django's ability to add to and retrieve from the database.

RESTful API

JSON

Our API is a RESTful API that performs transactions using JSON, a lightweight standard format that uses human readable text to transmit data objects consisting of attribute-value pairs. Although originally derived from JavaScript, its simplicity and power make it an extremely widespread web standard. We separated our HTML returns and our JSON returns by prepending /api. Example: API endpoint for

‘/people’ would be ‘/api/people/’ .

GET

Example snippets of JSON requests and replies are shown below for the Senator class. Similar requests and replies are analogous for the other classes.

/api/senators - GET

Gets all of the senators. Example:

```
+ Response 200 (application/json)


        [
            {
                "id": 1,
                "name": "Jane Nelson",
                "party": "Republican",
                "occupation": "Businesswoman, former teacher",
                "legistlative_experience": "Disaster Relief",
                "district": "12",
                "twitter": "https://twitter.com/SenJaneNelson",
                "facebook": "https://www.facebook.com/SenatorJaneNelson",
                "committees": [1,2]
            },
            {
                "id": 2,
                "name": "John Whitmire",
                "party": "Democratic",
                "occupation": "Attorney",
                "legistlative_experience": "",
                "district": "15",
                "twitter": "",
                "facebook": "",
                "committees": [2]
            }

        ]
```

/api/senators/{id} - GET

Gets one of the senators. Example:

```
+ Response 200 (application/json)
    + Body

          {
                "id": 1,
                "name": "Jane Nelson",
                "party": "Republican",
                "occupation": "Businesswoman, former teacher",
                "legistlative_experience": "Disaster Relief",
                "district": "12",
                "twitter": "https://twitter.com/SenJaneNelson",
                "facebook": "https://www.facebook.com/SenatorJaneNelson",
                "committees": [1,2]
            }
```

/api/senators/{id}/bills - GET

```
All of the Bills that this senator has authored
###List the bills [GET]
+ Response 200 (application/json)
        [
            {
                "id": 1,
                "name": "SB 63",
                "authors": [1],
                "legislative_session": "83(R)",
                "date_proposed": "11/12/2012",
                "date_signed": "6/14/2013",
                "date_effective": "6/14/2013",
                "status": "Signed into law",
                "url":
"http://www.legis.state.tx.us/BillLookup/History.aspx?LegSess=83R&Bill=SB63",
                "primary_committee": 1,
                "Description": "Relating to consent to the immunization of certain children.",
                "voters": [2, 3]
            }
        ]
```

Others

- /api/bills – GET – Gets a collection of all of the bills in the database
- /api/bills/{id} – GET – Gets a specific bill, where {id} is the identification key of the desired bill
- api/bills/{id}/senators – GET – Gets a list of senators and their corresponding votes for the bill with id {id}
- api/bills/{id}/authors – GET – Gets a list of the senators who authored the bill with id {id}
- /api/committees – GET – Gets a collection of all of the committees in the database
- /api/committees/{id} – GET – Gets a specific committee, where {id} is the identification key of

the desired committee

- /api/committees/{id}/senators – GET – Get all the senators in the committee with id {id}

- /api/committees/{id}/bills – GET – Get all the bills originating in the committee with id {id}

PUT

/api/senators/{id} - PUSH

Update an existing senator

```
+ Request (application/json)

        {
            "id": 1,
            "name": "Jane Nelson",
            "party": "Republican",
            "occupation": "Businesswoman, former teacher",
            "legistlative_experience": "Disaster Relief",
            "district": "12",
            "twitter": "https://twitter.com/SenJaneNelson",
            "facebook": "https://www.facebook.com/SenatorJaneNelson",
            "picture": "none",
            "committees": [1,2]
        }
+ Response 204
```

Others

- /api/bills/{id} – PUT – Updates a specific bill, where {id} is the identification key of the desired bill
- /api/committees/{id} – PUT – Updates a specific committee, where {id} is the identification key of the desired committee

POST

/api/senators/{id} - POST

Create a new senator entry

```
+ Request (application/json)

        [
            {
                "name": "Jane Nelson",
                "party": "Republican",
                "occupation": "Businesswoman, former teacher",
                "legistlative_experience": "Disaster Relief",
                "district": "12",
                "twitter": "https://twitter.com/SenJaneNelson",
                "facebook": "https://www.facebook.com/SenatorJaneNelson",
                "committees": [1,2]
            }
        ]

+ Response 201 (application/json)

        {
            "id": 1
        }
```

Others

- /api/bills – POST – Adds a new bill to the collection of bills
- /api/committees – POST – Adds a new committee to the collection of committees

DELETE

/api/senators/{id} – DELETE

```
+ Response 204
```

Others

- /api/bills/{id} – DELETE – Delete a bill from the database, where {id} is the identification key of the bill to remove
- /api/committees/{id} – DELETE – Delete a committee from the database, where {id} is the identification key of the committee to remove

API Unit Tests

Unit tests for the API, found in cs373-idb/tests.py, use the Python unittest library to individually test each endpoint and the associated HTTP methods. The idea is to comprehensively test the RESTful API on the live Heroku server. To do this, at the start of each test method we open a connection to the app on Heroku which we close at the end of the method. In each test, the response status of the HTTP request is checked. In the methods that return a response body, the body is in byte form and must be decoded to a string and then loaded into JSON using the "json.loads" function.

In the tests for the POST and PUT HTTP methods, the variable "values" is used with the "json.dumps" function to convert a python list/dictionary into a JSON object to be passed to the client. For the POST method, the response body is a JSON object containing only the id of the new object. For the PUT method, there is no response body check.

In the tests for the GET HTTP methods, the variable "desired_body" contains the expected response body in the form of a python object that is checked against the response returned from the client.

In the tests for the DELETE HTTP methods, there is no "desired_body" or "values" variables as no JSON objects are passed to or from the client.

Frontend Design

Overview

The driving idea behind the UI design of the Texas Law Database is clarity. We believe that clear, readily available information best serve our goals for not only a functional, usable website, but also serve our goal to presenting information to voters in an easily accessible way. We believe that an open democracy is best served when citizens have full access to information, and the best way to achieve that is to not hide information, but to display it in the open. To that end, we have, wherever possible, eschewed complicated menus and hierarchal displays of data and instead opted for a clear, concise view of information. Furthermore, we recognize that our website is a tool for the 21st century voter, and with that comes the expectation that our website looks modern, feels modern, and is usable on modern devices, such as phones and tablets. To that end, we have used Twitter Bootstrap as our front end framework. Not only does Twitter Bootstrap provide a sleek, intuitive, powerful framework for our frontend, it also provides features such as responsive CSS to allow our website to be used by a variety of devices at a variety of resolutions. With these goals in mind, the following pages will attempt to provide a user perspective on the frontend design of the Texas Law Database

Splash Page

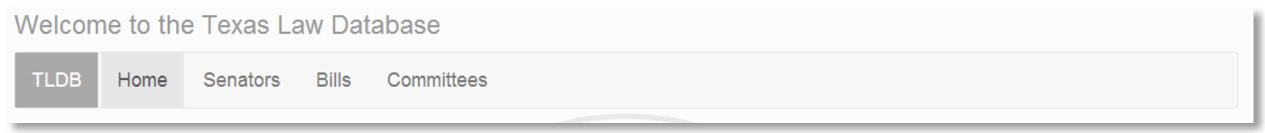


FIGURE 4: DESKTOP SPLASH PAGE HEADER

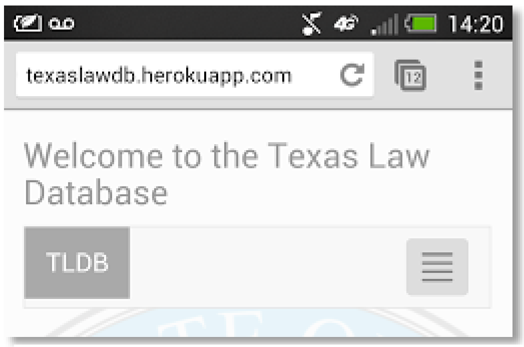


FIGURE 2: SPLASH HEADER MOBILE UNENGAGED

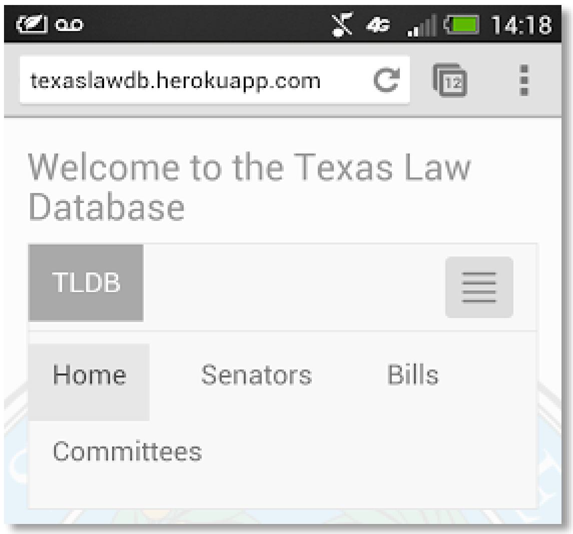


FIGURE 3: SPLASH HEADER MOBILE ENGAGED

The first thing a user sees when navigating to the Texas Law Database is the splash page. There, as required, one can see a list of contributors to the project. More importantly, one can see the header, which lists the categories for which the Texas Law Database provides information about, each of which is represented by a Django model. This allows the user to easily and quickly select what they are immediately interested in, allowing easy navigation to the site's full capabilities. This header is placed on every page of the Texas Law Database, so that a user can return to the splash page or navigate to any category from any page.

Here already, we can see the power of Twitter Bootstrap. The header is preserved in function on mobile devices, and we see that the menu collapses into a touchable element. In this way, mobile users can still have the full functionality of the header without sacrificing valuable mobile screen real estate.

Senators View

Overview Page

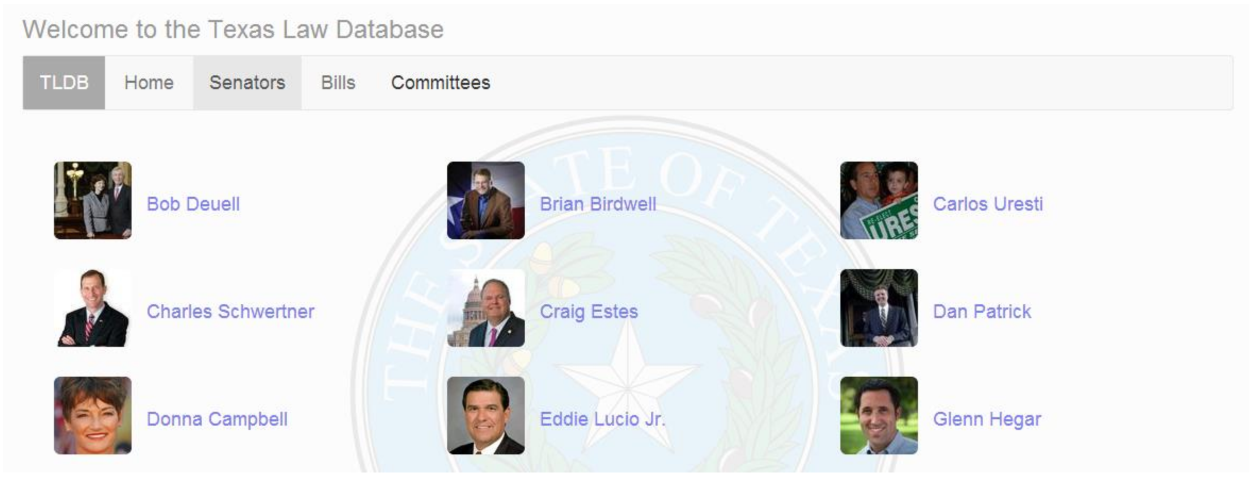
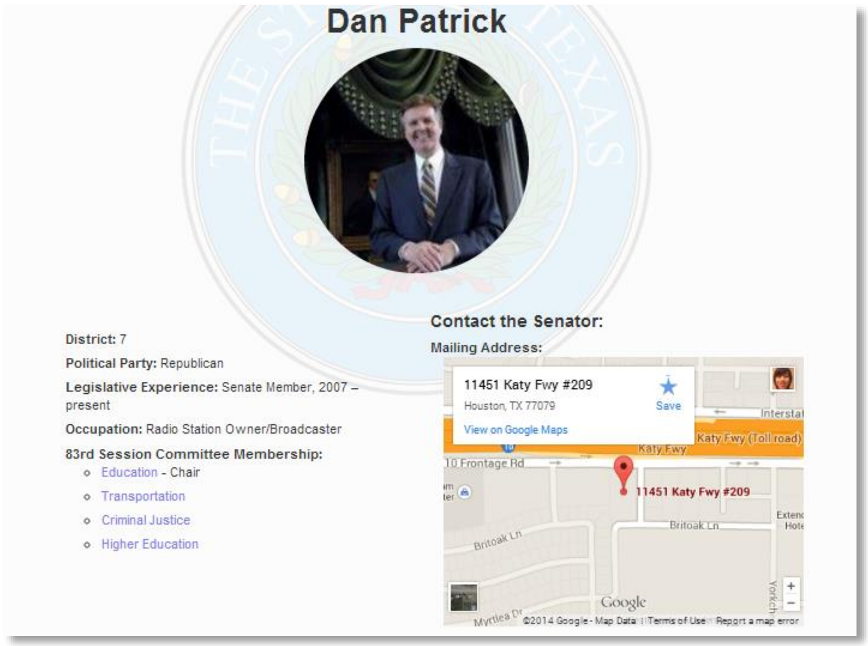


FIGURE 5: SENATOR OVERVIEW PAGE

Navigating to the senators page (at texaslawdb.heroku.com/senators or from clicking on the senators link in the header) brings us to the senator overview page. Here, we list the members of the Texas Senate of the current legislature and display their picture, taken from their Twitter profile. Not only is this source of pictures programmatically easier, this allows voters to visually search for their senator in a way that they are already used to through twitter. This way, voters already familiar with their senator can easily, through recognition of familiar visual cues, navigate to the page of the senator they're interested in.

Individual Page


Dan Patrick

District: 7

Political Party: Republican

Legislative Experience: Senate Member, 2007 – present

Occupation: Radio Station Owner/Broadcaster

83rd Session Committee Membership:

- Education - Chair
- Transportation
- Criminal Justice
- Higher Education

Contact the Senator:

Mailing Address:

11451 Katy Fwy #209
Houston, TX 77079

FIGURE 6: SENATOR ATTRIBUTE VIEW

Navigating to an individual senator's page from the overview page brings us to this view. Here, we can see an individual senator's district, party, legislative experience, occupation, committee membership, and address. These attributes are all contained in the Django model, and provide valuable information to voters interested in a legislator. The committee membership in particular is modeled entirely through Django, and links to those committees from a senator's individual page will bring you to the individual page for the committee.

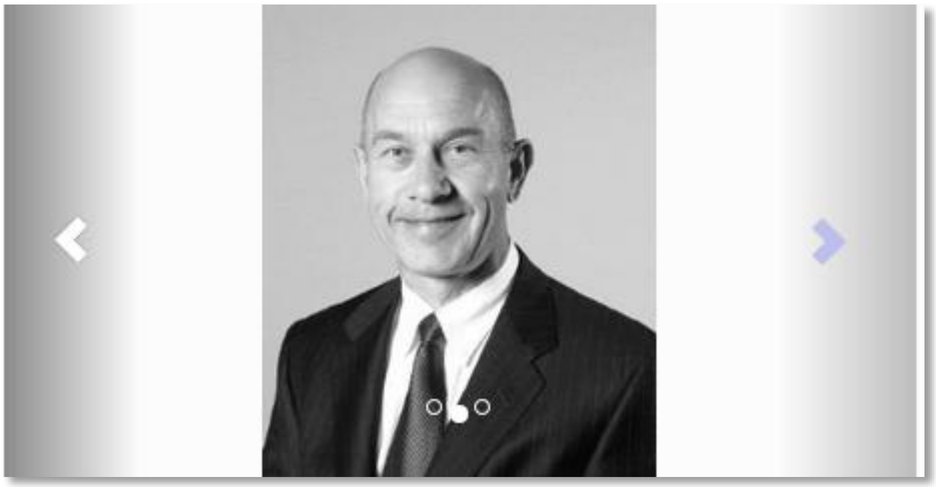
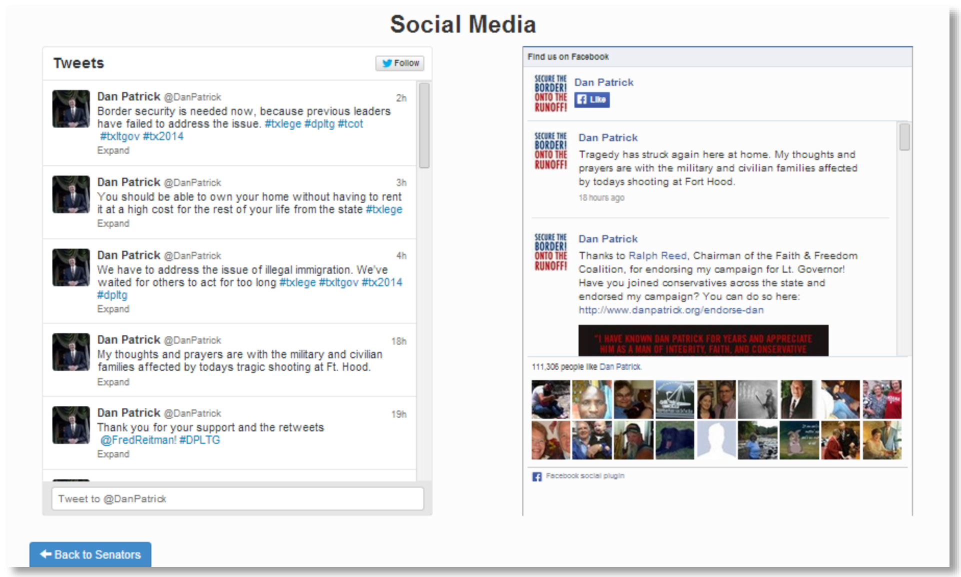


FIGURE 7: SENATOR PHOTO CAROUSEL

Underneath the senator's information, we have a carousel of photos for each senator. This allows voters to connect further with their legislators, as seeing them in different situations, ones not necessarily picked by the legislators themselves as in their Twitter pictures, allows voters to more deeply connect with their legislators. This is modeled by our photo model in the Django database.



Social Media

Tweets

Follow

Dan Patrick @DanPatrick 2h
Border security is needed now, because previous leaders have failed to address the issue. #txlege #dpltg #tcot #txltgov #tx2014
Expand

Dan Patrick @DanPatrick 3h
You should be able to own your home without having to rent it at a high cost for the rest of your life from the state #txlege
Expand

Dan Patrick @DanPatrick 4h
We have to address the issue of illegal immigration. We've waited for others to act for too long #txlege #txltgov #tx2014 #dpltg
Expand

Dan Patrick @DanPatrick 18h
My thoughts and prayers are with the military and civilian families affected by todays tragic shooting at Ft. Hood.
Expand

Dan Patrick @DanPatrick 19h
Thank you for your support and the retweets @FredReitman! #DPLTG
Expand

Tweet to @DanPatrick

Find us on Facebook

Dan Patrick
Like

Dan Patrick
Tragedy has struck again here at home. My thoughts and prayers are with the military and civilian families affected by todays shooting at Fort Hood.
18 hours ago

Dan Patrick
Thanks to Ralph Reed, Chairman of the Faith & Freedom Coalition, for endorsing my campaign for Lt. Governor! Have you joined conservatives across the state and endorsed my campaign? You can do so here: http://www.danpatrick.org/endorse-dan

"I HAVE KNOWN DAN PATRICK FOR YEARS AND APPRECIATE HIM AS A MAN OF INTEGRITY, FAITH, AND CONSERVATIVE

111,306 people like Dan Patrick.

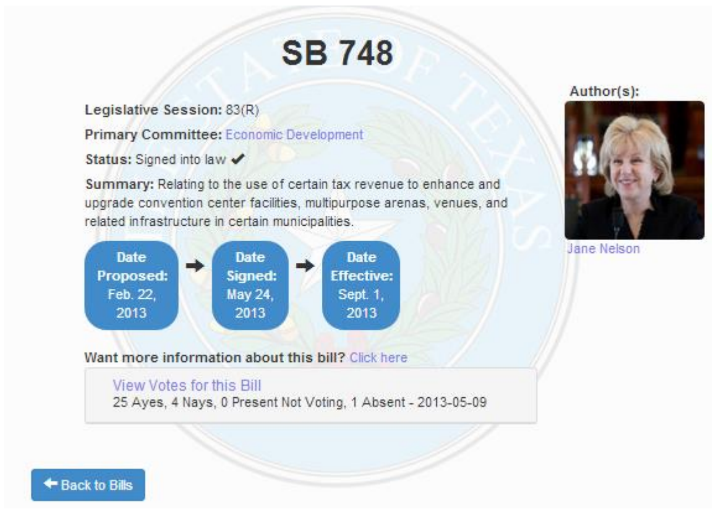
Facebook social plugin

Back to Senators

Figure 8: Senator Social Media View

Scrolling down further brings us to the social media feeds of the legislators. We believe that there is no better tool in the 21st century for voters to interact with their legislators than through social media. An open social media platform allows voters to connect with legislators on an unprecedented basis. In many cases, legislators themselves will run their own social media operations, allowing voters ready access to their legislators' thoughts and actions in a nearly unrestricted way. Since social media is so important to our goal of furthering an open-source democracy, we have given it great prominence on our senators' pages. We hope that voters can use our website to learn about their legislators' actions from us, and interact with them without having to leave our website. This way, we give users all the tools they need to interact with their legislators, and open up democracy beyond what information they can get from traditional media sources.

Bills



SB 748

Legislative Session: 83(R)
Primary Committee: Economic Development
Status: Signed into law ✓
Summary: Relating to the use of certain tax revenue to enhance and upgrade convention center facilities, multipurpose arenas, venues, and related infrastructure in certain municipalities.

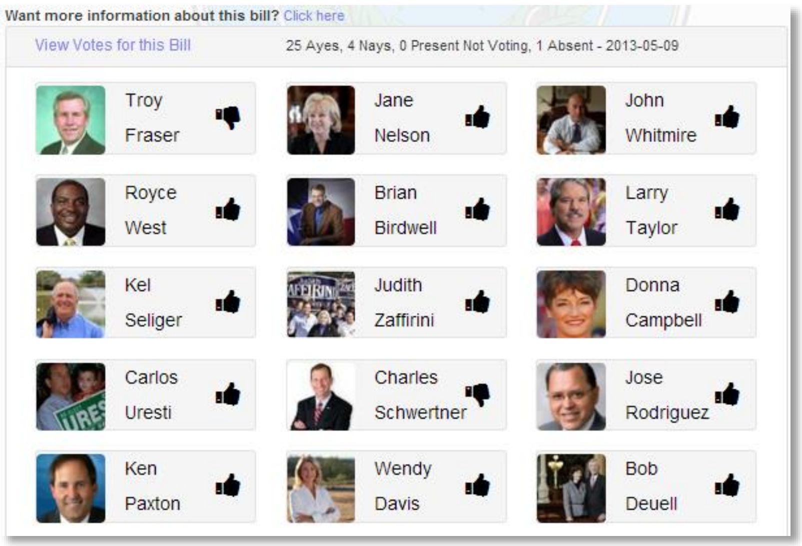
Date Proposed: Feb. 22, 2013 → **Date Signed:** May 24, 2013 → **Date Effective:** Sept. 1, 2013

Want more information about this bill? Click here

View Votes for this Bill
25 Ayes, 4 Nays, 0 Present Not Voting, 1 Absent - 2013-05-09

FIGURE 9: INDIVIDUAL BILL VIEW

From the bill view, voters can learn about individual bills. They can see the legislative session to which the bill belongs to, the primary committee through which the bill was passed, the status of the bill, dates regarding the bill, the author to the bill, and a link to the full text of the bill. Of these, the committee and author are modeled through Django, and clicking on those links will bring you to the appropriate page in the Texas Law Database. Furthermore, one can see a vote summary for the bill, which summarizes the votes, as well as displaying the date of the vote.



Want more information about this bill? Click here

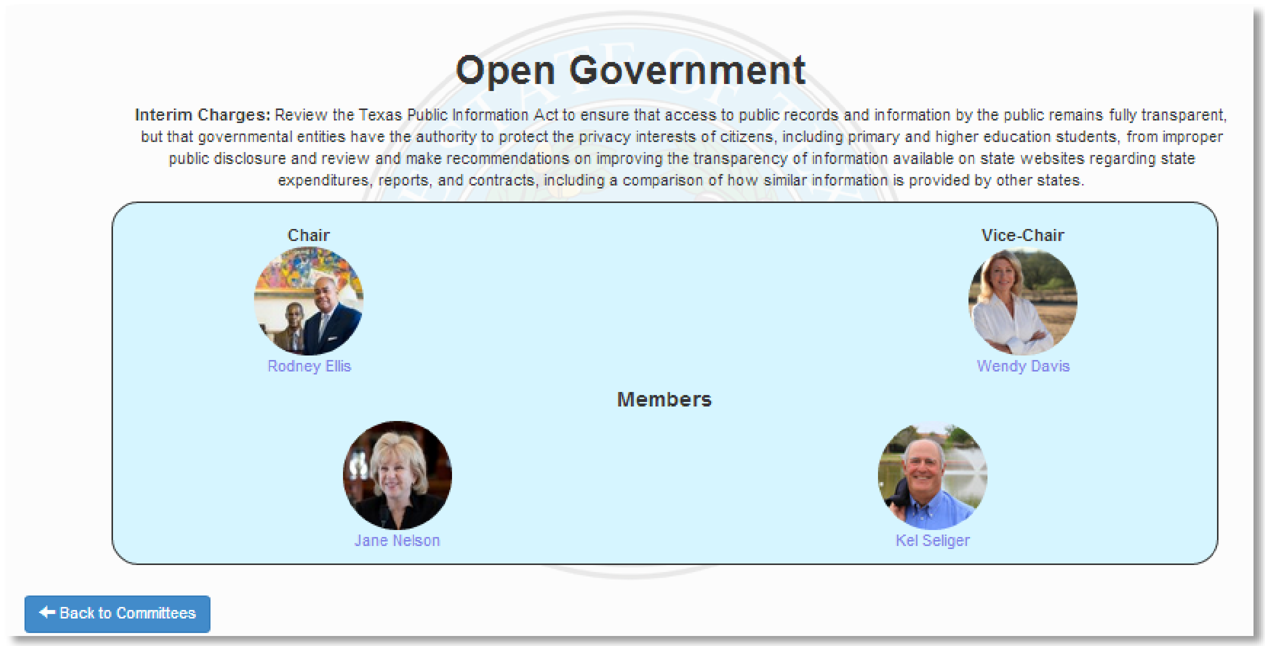
View Votes for this Bill — 25 Ayes, 4 Nays, 0 Present Not Voting, 1 Absent - 2013-05-09

Troy Fraser 👎	Jane Nelson 👍	John Whitmire 👍
Royce West 👍	Brian Birdwell 👍	Larry Taylor 👍
Kel Seliger 👍	Judith Zaffirini 👍	Donna Campbell 👍
Carlos Uresti 👍	Charles Schwertner 👎	Jose Rodriguez 👍
Ken Paxton 👍	Wendy Davis 👍	Bob Deuell 👍

FIGURE 10: BILL VOTE BREAKDOWN

Clicking on this vote summary will display the vote breakdown pane, where one can see a detailed graphical presentation of the bill's votes. From here, one can click on the name of each of the senators to navigate to the senator's page in the Texas Law Database, where they can learn more detailed information for each senator. Each of these votes is modeled by the vote model, and each senator is modeled through the Senator model.

Committees


Open Government

Interim Charges: Review the Texas Public Information Act to ensure that access to public records and information by the public remains fully transparent, but that governmental entities have the authority to protect the privacy interests of citizens, including primary and higher education students, from improper public disclosure and review and make recommendations on improving the transparency of information available on state websites regarding state expenditures, reports, and contracts, including a comparison of how similar information is provided by other states.

FIGURE 11: INDIVIDUAL COMMITTEE VIEW

Clicking on an individual committee will bring us to this view, where users can learn about the charge of a committee, and see a list of its members. These members are all senators as modeled by our senators model, and clicking on a link to a senator will bring us to the individual page for the senator.

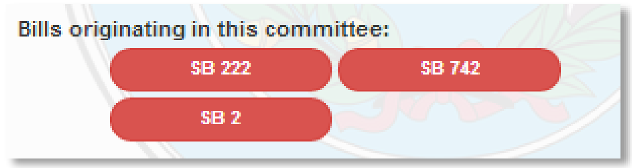


FIGURE 12: COMMITTEE BILL VIEW

Furthermore, for committees with associated bills in the database, we display a list of bills in the database that have gone through the committee. Here, users can click on the bills to navigate to the page for the bill in the Texas Law Database, where they can learn more detailed information about the bill.