

# Use Cases and Functional Requirements

PolitiMap Team

October 7, 2016

# PolitiMap

Some Local Bill

Some State Bill

Some Federal Bill

Some Local Bill

Some Local Bill

Some State Bill

Some Local Bill



# PolitiMap

Some Federal Bill



# Some Federal Bill

## Summary

This bill requires all citizens to pronounce the United States of America as "Murica".

## Supporting Representatives



## Associated Links

[Official Documentation](#)

[Discussion Board](#)

[Other Links](#)



Use Case Name:	View Federal Bills
Actors:	The Application User; The Swift iOS Application; The Backend Server
Description:	The application shall display a list of federal bills and further information about each upon selection.
Preconditions:	<ol style="list-style-type: none"><li>1. The device is connected to the internet.</li><li>2. The backend is loaded with the organized data.</li><li>3. The device has the application downloaded, updated, and open to the home screen.</li><li>4. The backend server is running, whether it be an AWS component or EC2 instance.</li></ol>

Postconditions:

1. A list of federal bills will be shown and likely scrolled-down, or the details of a specific bill will be showing.
2. A request will have been made to the AWS component or EC2 instance.
3. TBD - The federal bills may be cached in the user's device.

## Normal Flow:

1. The user chooses to view the list of bills relating to their local, state, and federal governments.
2. A GET request is made to the web server while the user sees a loading animation.
3. The user selects to filter the list of bills to show only federal bills.
4. The user scrolls through the bills and notices an interesting one.
5. The interesting bill is selected and the user reads related information, including a summary, associated representatives, and links to related external documentation.

## Alternative Flows:

1. The user selects a federal bills list instead of filtering a combined list.
2. The bills are loaded upon opening the application and only refreshed manually.
3. The bill information does not include associated representatives.
4. The bill shows dynamic data, such as votes and comments.

Exceptions:

1. Unable to load the bills through the lack of an internet connection.

Includes:

1. An external library to make the GET request.

Priority:

High - One of the core functions of the application.

Frequency of Use:

Depends on the number of users and other variables.

Special Requirements:

1. The application requests the data asynchronously.
2. The device is an iPhone 5 or newer.
3. Specific bill information is loaded with a GET request after a bill is selected.

# Functional Requirements

1. The system shall store the frequently accessed data, such as the list of federal bills, in AWS S3, to minimize the cost to \$.004 per 10,000 requests.
2. The system shall refresh the list of federal bills from GET requests through a network connection.
3. The system shall default to sorting bills by the timestamp of their last update in content.

## Non-Functional Requirements

1. The system shall load the list of bills in under two seconds.
2. The system shall have zero memory leaks.
3. Federal bills shall be updated within one day of receiving new information.

Use Case Name: Display Agenda  
Actors: The PolitiMap App  
Description: Display the agenda for the next City Council meeting  
Preconditions:

1. The user has chosen a location which has a city council
2. The location is currently supported by the app
3. The app is open
4. The user has chosen “City Council Meeting” from the list of events.

Postconditions:

1. The app shows the agenda for the next City Council meeting for the selected location.

Normal Flow:

1. The user opens app
2. The app requests data from server
3. (Alternative flow 1)
4. The app displays a list of upcoming events
5. The user chooses the City Council item

Alternative Flows: Alternative flow 1

1. Read data from cache

Priority: High

Assumptions: Council meeting agendas will be available for all supported locations

# Functional Requirements

1. The system shall load data in the JSON format specified in an appendix.
2. The system shall fetch the data to load from the web.
3. The system shall only load JSON data for the locations selected.

## Non-Functional Requirements

1. The system shall provide a button to add an event to the user's calendar.
2. The system shall display a list of upcoming meetings.
3. The system shall display the date and time of each meeting in the list of meetings.
4. The system shall display the location of each meeting in the list of meetings.

Use Case Name: Save Locations

Actors: The PolitiMap App

Description: Save multiple locations on the home page of the Application

Preconditions:

1. The location is currently supported by the app
2. The app is open
3. The app is on the home page

Postconditions:

1. The app shows the saved location on the home page.
2. The app should be able to access the location and show its Political information when tapped

Normal Flow:

1. The user opens app
2. The app loads locations saved by the user
3. The user views saved locations
4. (Alternative flow 1)
5. The user searches a new locations
6. The user chooses to save the location

Alternative Flows: Alternative flow 1

1. Save location

Priority:

High

Assumptions:

All locations can be found

# Functional Requirements

1. The system shall Alter order of saved addresses.
2. The system shall Delete addresses the user doesn't want anymore.
3. The system shall Edit saved addresses.

# Non-Functional Requirements

1. The system shall securely store the saved addresses.
2. Memory should be available to store the local addresses
3. Home page should be loaded instantly with the static information

Use Case Name: View bills that affect foreign students

Actors:

1. iOS Application
2. The back-end server

Preconditions:

1. The device is connected to the internet and the app is downloaded.
2. The app is opened showing the main page.

Postconditions:

1. One of the menus should direct the user to a new page.
2. The page will contain bills that are related to foreign students.

# Functional Requirements

1. The system will have a drop-down menu to display Bills related to foreign students.
2. The system will save recently viewed Bills.
3. The system will allow the users to call their city council by pressing the contact button.

# Non-Functional Requirements

1. UI is easy to follow for the users.
2. Pages are loaded in less than 2 seconds
3. The system will show a loading bar when pages are loading.

Use Case Name: Display Policy Positions  
Actors: The PolitiMap App  
Description: View policy summary of given local politician  
on active issues

Preconditions:

1. The user's location is currently supported by the app
2. The user has chosen the city council member to evaluate
3. Push
4. The user has selected "Policy Positions" from menu options

Postconditions:

1. The app displays a summarized list of policy positions on active local legislation, and lists a "position not available" message if position is indeterminate

## Normal Flow:

1. The user opens the app
2. The user specifies their location, or a cached location is processed
3. The user selects "Politicians" from the lower UI menu
4. The user selects their chosen politician from the "Your Representatives" list
5. The user navigates to "Policy Positions" using the UI menu

Priority:

Medium

Special Requirements:

None

Assumptions:

Policy positions will be available publicly through government or council member's website

# Functional Requirements

1. The system shall load data in the JSON format specified in an appendix.
2. The system shall fetch the data using AJAX or other method of web request.
3. The system shall only load JSON data for the locations selected by the user.
4. The system shall data on politicians must be verifiable and completely accurate.

# Non-Functional Requirements

1. The system shall provide menu UI on politician page for ease of navigation.
2. The system shall display bulleted list of policy positions on active legislation.