CMSC 447

Software Design Description (SDD)

Name	Role	Signature
Brett Smith	Point of Contact, Documentation	
Brian To	Documentation	
Kristina Bridgwater	Front End Development	
Denise Howell	Not currently assigned	
Eric Hale	Not currently assigned	
Daniel Howard	Not currently assigned	

1 Scope

This section shall be divided into the following paragraphs.

1.1 Identification

The purpose of this document is to give a detailed description as well as an overview of how the T3AM Gerrymandering Tool application will be built. This document will provide graphical models and diagrams to aid the development of the CSCI.

1.2 System overview

The “T3AM Gerrymandering Tool” is a web based application that generates a state map with Gerrymandered congressional districts based on the user’s input, with the available inputs being current distribution, Republican, Democratic, or swing. Additionally, the application displays district information such as political party majorities and percentages. This software is for Costas Likakis, project sponsor.

1.3 Document overview

This document will be divided into seven sections with various subsections. Included chapters are:

- Scope
    - Gives the general overview of the SDD.
- Referenced Documents
    - Includes any documents referenced by the SDD.
- Design Decisions
    - Section regarding the CSCI’s behavioral design, including how it will behave and other decisions regarding the CSCI’s design.
- Architectural Design
    - Describes the CSCI architectural design, the design depends upon system states or modes.
- Detailed Design
    - Describes each unit of the CSCI.
- Requirements traceability
    - Matches each requirement to the origin.
- Additional notes
    - Notes and raw data.

2 Referenced documents

Possible inspirations for gerrymandering algorithm:

http://rangevoting.org/GerryExamples.html

http://web.mit.edu/~jhurwitz/www/census/SIURO.pdf http://web.cs.swarthmore.edu/~adanner/cs97/s08/papers/benn_german.pdf

3 CSCI-wide design decisions

a. Design decisions regarding inputs the CSCI will accept and outputs it will produce, including interfaces with other systems.
    - The landing page shall have 3 inputs from the user:
        - The Google Captcha
        - The “agree to Terms of Service” checkbox.
        - The “continue to site” button.
    - The landing page shall have 1 output to the user:
        - Upon failing the Captcha, the Captcha will prompt the user again.
    - The main page will have 6 inputs from the user:
        - Button input from the user:
            - Current distribution
            - Democratic distribution
            - Republican distribution
            - Swing distribution
        - “State drop down menu” input
        - “On-hover” map input
        - “View Country” input
    - The main page will have 3 outputs:
        - A map to display to user
        - Sidebar information
        - “On-hover” map information

b. Design decisions on CSCI behavior in response to each input or condition, including actions the CSCI will perform, response times and other performance characteristics, description of physical systems modeled, selected equations/algorithms/rules, and handling of unallowed inputs or conditions.
    - On the following user inputs:
        - On mouse hover over map:
            - Identify which part of the map the user is hovering over.
            - Get information from the database.
            - Display said information on a tooltip.
        - On button click:
            - Identify which button was pushed.
            - Generate a map based on the button:
                - Get information from the database.
                - Input information from the database.
                - Generate updated map.
            - Display updated map.

c. Design decisions on how databases/data files will appear to the user.

- Database information will appear in the form of:
    - Sidebar information
    - “On-hover” information
    - A state map

d. Selected approach to meeting safety, security, and privacy requirements.
   - The site shall be protected from bots via Captcha.
   - Database is hidden from the user, and cannot be accessed directly from the application.

4 CSCI architectural design

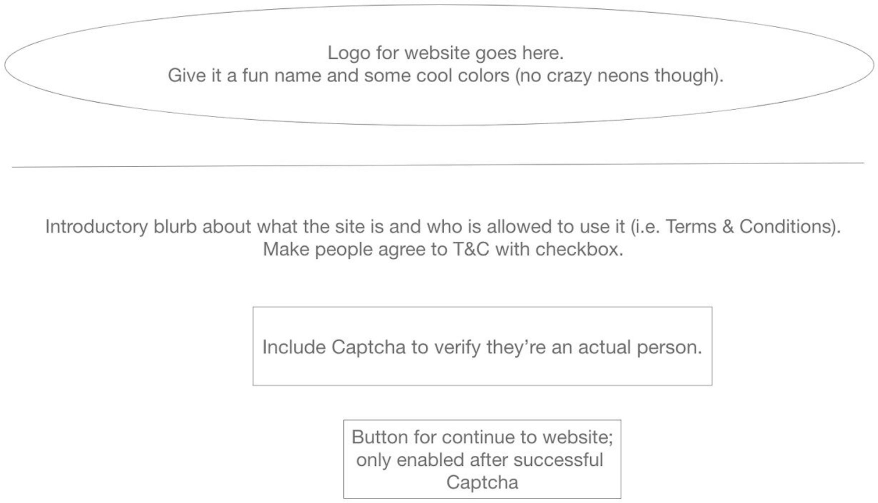
4.1 CSCI components

a. The database shall consist of the necessary information about each district for each possible state.
   - ○ The purpose of this database is to hold the information about the current distributions of each district, so the application has a baseline to go off of when the user wants to change distributions.
   - ○ Development status/type: New development
       - This data will need to be gathered from one or more reliable sources and inserted into the database by our team.

b. The landing page shall consist of:
   - Checkbox to ensure the user agrees to Terms and Conditions.
   - Captcha verification module.
   - Button to continue to the main page of application.
   - ○ The purpose of the landing page is to give users vital background information and a description about the purpose of the application.
   - ○ Development status/type: New development
       - This page will need to be created by our team but will use some existing designs, such as the Captcha module.

c. The main page shall consist of:
   - The state map that displays the currently selected political distribution.
   - Buttons that allow the user to change distributions, as described in 3.a.
   - Sidebar that displays relevant information about a specific area of the map, selected by the user.
   - ○ The purpose of this page is to allow users to see the current political distributions of a state, or view hypothetical distributions based on changing one or more district’s political affiliations.
   - ○ Development status/type: New development
       - This page will need to be created by our team but will use some existing designs, such as the Google Maps API for the state map.
       - The map will utilize data contained in the database described in 4.1.a to display district distributions.

4.2 Concept of execution



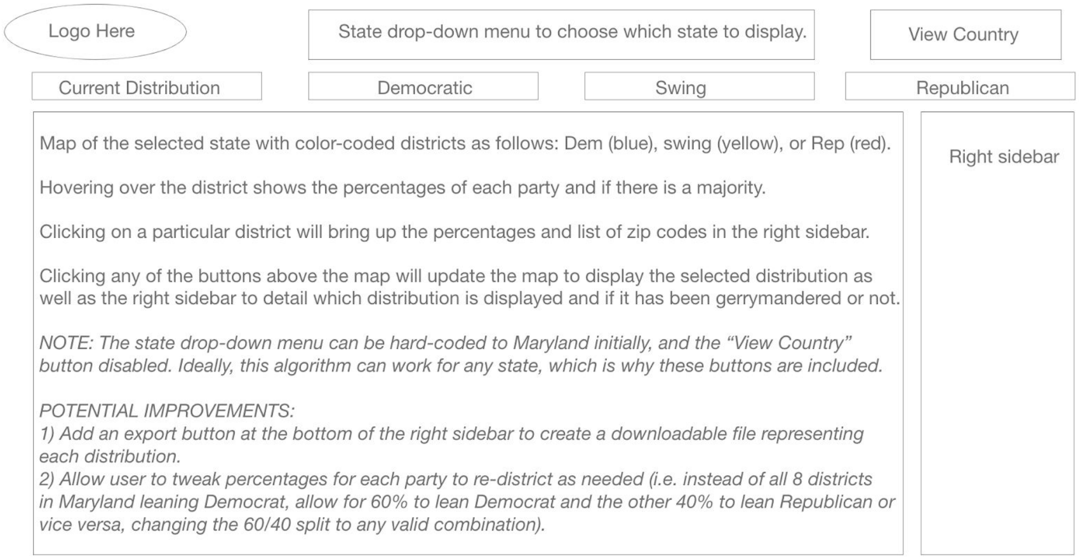
Logo for website goes here.
Give it a fun name and some cool colors (no crazy neons though).

Introductory blurb about what the site is and who is allowed to use it (i.e. Terms & Conditions).
Make people agree to T&C with checkbox.

Include Captcha to verify they're an actual person.

Button for continue to website; only enabled after successful Captcha

Figure 1. Landing page wireframe



Map of the selected state with color-coded districts as follows: Dem (blue), swing (yellow), or Rep (red).

Hovering over the district shows the percentages of each party and if there is a majority.

Clicking on a particular district will bring up the percentages and list of zip codes in the right sidebar.

Clicking any of the buttons above the map will update the map to display the selected distribution as well as the right sidebar to detail which distribution is displayed and if it has been gerrymandered or not.

NOTE: The state drop-down menu can be hard-coded to Maryland initially, and the "View Country" button disabled. Ideally, this algorithm can work for any state, which is why these buttons are included.

POTENTIAL IMPROVEMENTS:
1) Add an export button at the bottom of the right sidebar to create a downloadable file representing each distribution.
2) Allow user to tweak percentages for each party to re-district as needed (i.e. instead of all 8 districts in Maryland leaning Democrat, allow for 60% to lean Democrat and the other 40% to lean Republican or vice versa, changing the 60/40 split to any valid combination).

Figure 2. Main page wireframe

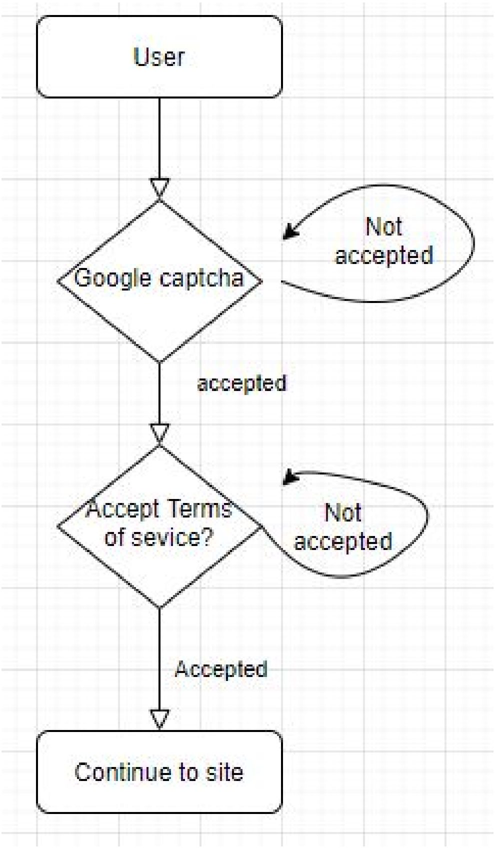


Figure 3. Main page access

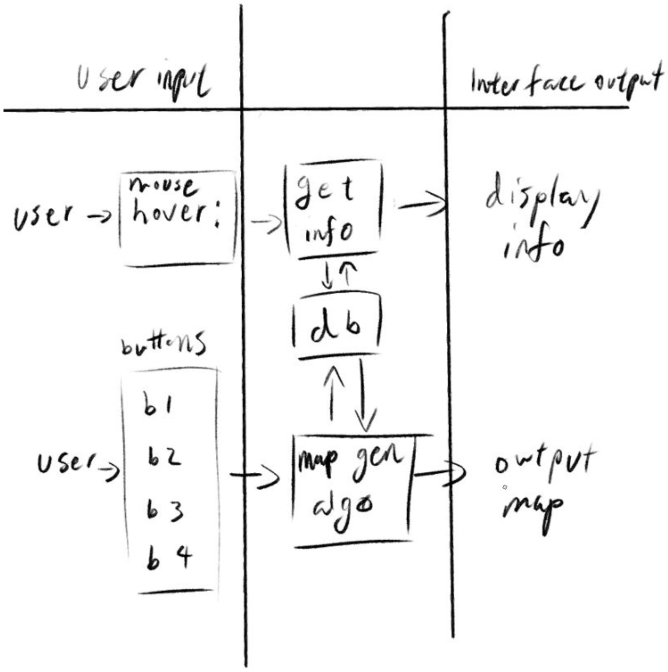


Figure 4. Input output table

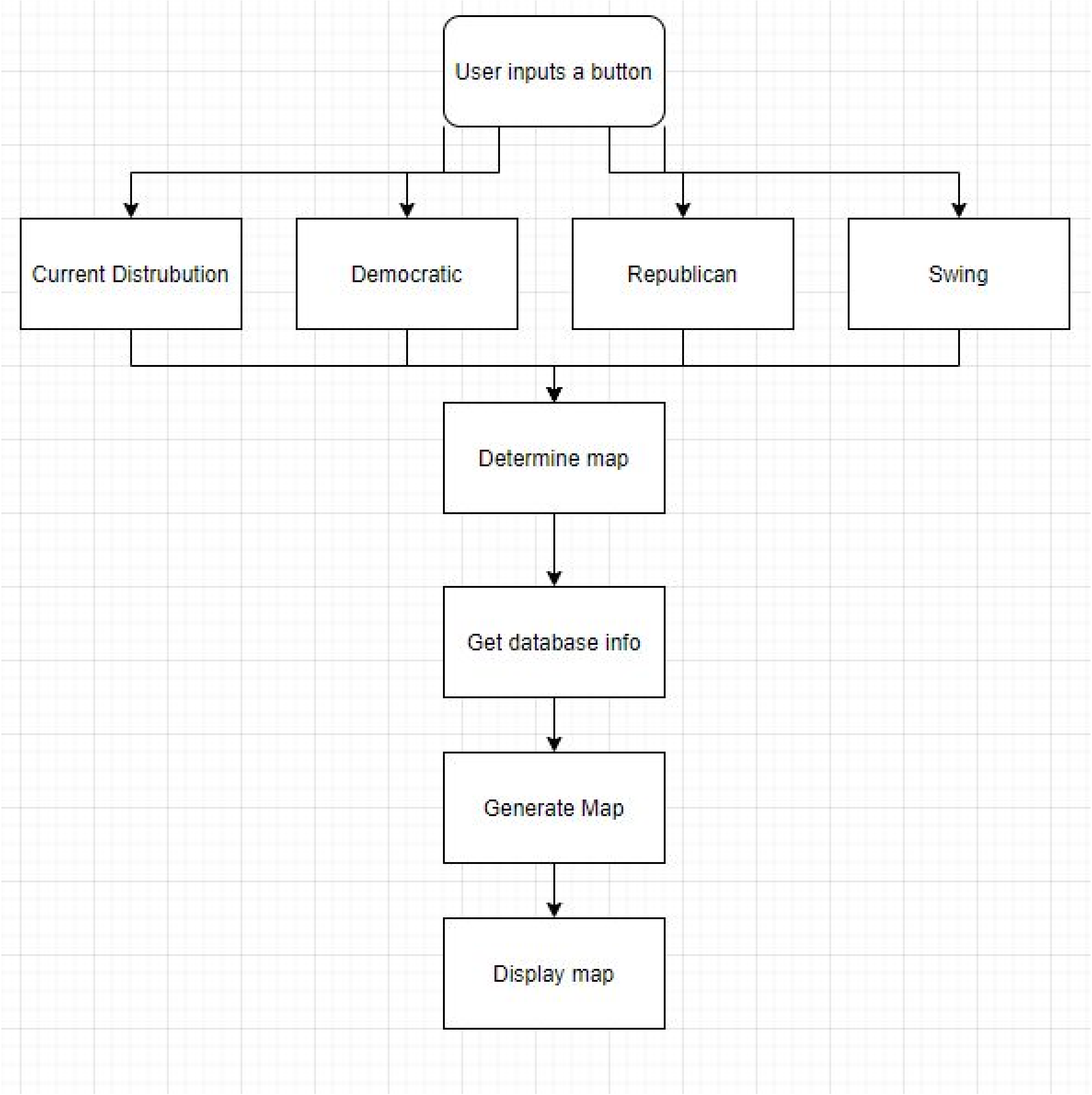


Figure 5. Map generation

4.3 Interface design

4.3.1 Interface identification and diagrams

All essential interfaces have been identified in 4.1 and diagrams of landing page and main page exist in 4.2.

4.3.2 Database

a. Priority Assigned: #1 priority
b. Type of interface: Web page with storage/retrieval of data
c. Characteristics of data elements:

        i. Percentages of distributions represented as integers/decimals.
        ii. Population sizes represented as integers.
    d. The database will be accessed by the main page as a part of its functionality.

4.3.3 Main Page

a. Priority Assigned: #2 priority
b. Type of interface: Real-time retrieval/modification of data
c. Characteristics of data elements:
    i. Percentages of distributions represented as integers/decimals.
    ii. Population size represented as integers.
d. The map on the main page will access data in the created database (described in 4.1.a).

4.3.4 Landing Page

a. Priority Assigned: #3 priority
b. Type of interface: Web page that provides description of application
c. Characteristics of data elements:
    i. Text that provides description will be represented as strings.
d. Once the “continue” button is clicked on this page, access to the main page will be provided.

5 CSCI detailed design

5.1 Main Page

a. Unit design decisions:
    i. The map featured on this page shall utilize an algorithm to determine what data mappings will create the desired distribution of the user.
        1. Final algorithm for this procedure has not yet been decided on, but might be influenced by the ones in the Referenced Documents section of this document.
b. Constraints, Limitations, Unusual Features:
    i. Realistically, this map feature should work for any U.S state, but for the purpose of this project, only Maryland has been specified as a required state.
c. The programming language to be used and rationale for its use if other than the specified CSCI language:
    i. Python will be used as the programming language for it’s ease of use with Flask.
d. Software unit contains, receives, or outputs data, a description of its inputs, outputs, and other data elements and data element assemblies, as applicable.
    i. The software unit shall receive button inputs from the user, which determine which map to output.
    ii. The software unit shall receive data from various databases (to be determined).
        1. Voting distribution
        2. Map information such as district lines, zip codes
    iii. The software unit shall output a map based on the input given by the user.
    iv. The sidebar shall display map information based on the map.
        1. Voting distribution

2. Map information such as district lines, zip codes

v. Logic to be used by the software unit:

1. Refer to figure 4
2. Refer to figure 5

6 Requirements traceability

This section defines where each software unit requirement came from.

4.3.2 Database - Requested by Customer

4.3.3 Main Page - Requested by Customer

4.3.4 Landing Page - Requested by Customer

7 Notes

CSCI - (Computer Software Configuration Item) A group of software treated as a single entity.

“On-Hover”: This is referred to several times in the document. This simply refers to when the user hovers their mouse over a section of the state map and information/statistics will be displayed for that section of the map.