Requirements

## For SOTU-db

By Tyler Monaghan

### Assisted by

Abdur Khan

Tina Figueroa

# Contents

For SOTU-db 1

Assisted by 1

Contents 2

I. Introduction 3

Purpose 3

Scope 3

Definitions, acronyms, and abbreviations 3

References 3

Overview 4

II. Overall Description 4

Product Perspective 4

System Interfaces: 4

User Interfaces: 4

Hardware Interfaces 4

Software interfaces 4

Communications Interfaces 4

Memory Constraints 4

Operations 4

Product Functions 4

User characteristics 5

Constraints 5

Assumptions and Dependencies 5

Apportioning of Requirements 5

III. Specific Requirements 5

External Interfaces 5

Functions 5

The main screen 5

The text search box and the EXPLORE button 5

NAME landing page 6

YEAR landing page 6

TEXT landing page 6

Info Cards 7

Main menu 7

About screen 7

Performance Requirements 7

Database Requirements 7

Design constraints 8

Standards compliance 8

Software system attributes 8

IV. Appendices 9

Appendix X: References 9

# Introduction

## Purpose

The purpose of this SRS (system requirements specifications) document is to outline the key functions and requirements for the SOTU-db software. This requirements document is intended for Dr. George Thiruvathukal, DIGH-402 Professor and grader of this document. The development of this document will also help guide the overall direction of the project.

## Scope

1. This SRS is for the software application “SOTU-db,” also known as State of the Union Database.
2. The software will:

* Accept user text string queries
* Allow users to read, search, and compare text of different State of the Union (SOTU) addresses
* Provide “info cards” containing interesting insights or suggestions for navigating through the software

1. The software will not:

* Allow users to login, save their work, or track progress

1. This software is intended for a range of users (see “Personas” document) from high school students working on an assignment to professional journalists gathering data for publication in an article. The software emphasizes accuracy, ease-of-use, and playful interaction as facilitating active user engagement is a key goal of this project.

## Definitions, acronyms, and abbreviations

**SOTU-db:** The name of the project, unofficially standing for “State of the Union Database.”

**SOTU:** “State of the Union,” also used as shorthand to describe any president’s annual address included in this database.

**SRS:** system requirements specifications; this document

**Info card**: an interactive design element that displays information for users to dismiss or click/tap to explore further

**Database:** used to describe the collection of objects that make up the project documents.

**Object:** the textual documents in the “database.” Each file containing a copy of an address/speech is an “object.”

**Corpus:** the selection of documents that are designated as the primary versions of each annual address; the corpus is the main body of text for search, analysis, visualizations, etc.

## References

See Appendix X for references.

## Overview

The rest of the SRS is organized in accordance with the IEEE “recommended practice for software engineering requirements.” I will probably replace this with a ToC, this seems silly to me.

# Overall Description

This section explains the background for the product requirements that are stated later in this document.

## Product Perspective

This product is designed to be fully independent and self-contained within the following constraints:

* A web version of the software should be accessible through any current standards-compliant web browser. The software will be developed for and tested in Chrome.
* An android version of the software will be available as an application package (.apk) file and should be accessible on any current version of Android operating system

### System Interfaces:

Web and Android application versions.

### User Interfaces:

On mobile devices (web or Android application) SOTU-db will be designed to display properly on a portrait-oriented device (640-720px wide, variable height). As much as possible, universal and accessible design principles will be used to ensure the software maintains maximum compatibility over time and across a range of users. On a mobile device the interface only requires access to a keyboard and the ability to tap the screen. On a web interface, only a mouse and keyboard are required.

### Hardware Interfaces

*Specify the logicalq characteristics of each interface between software product and hardware product*

None to speak of, this product will run on current versions of Chrome and Android operating systems.

### Software interfaces

None required, this product is self-contained and standalone.

### Communications Interfaces

Internet access is required to operate the software. Network access permission required on Android.

### Memory Constraints

None

### Operations

The software conducts retrieval and analysis for the user based on user queries. Therefore the software has no unattended operations.

## Product Functions

See draw.io sketch.

## User characteristics

See personas document. In general:

* User technical skill ranges from daily power-user to casual smartphone user
* User content knowledge ranges from a neophyte student given an assignment to a professional exploring symbolic language for their career’s work
* General education levels range from high school with significant gaps to professional degree-holder

## Constraints

Must have a mobile/low-data/GO version of the app for users without reliable internet connections. Must be accessible on a range of device types.

## Assumptions and Dependencies

This SRS assumes that Chrome and Android OS maintain stable, standards-compliant releases (currently Chrome 65 and Android 7.1.1). These will be required for the software to function.

## Apportioning of Requirements

Features delayed for future releases:

* Interactivity / social commenting / user contributions
* Social / crowdsourced transcriptioning

# Specific Requirements

## External Interfaces

Users will access sotu-db by navigating to a specific address in a web browser, such as Chrome, or by opening the Android app.

## Functions

### The main screen

*accessible via app start or web browser visit*

At the main screen, users have three main options. 1) use the text search box and the “EXPLORE” button; 2) use the info cards at the bottom of the screen to engage in a guided search, 3) use the menu or the “about SOTU-db” link to navigate to documentation.

### The text search box and the EXPLORE button

*accessible via main screen*

A user may type in a search term, phrase, name, or year into the text box on the main screen. The software will determine whether a name or year has been entered into the box and direct users to the appropriate name landing page or year landing page. Otherwise, the software will try to direct a user to the appropriate text landing page.

### NAME landing page

*access by typing a recognized president’s name into search box*

This landing page would feature various info cards that would highlight and illustrate facts about this president’s speeches. For example:

* a card showing what years that president delivered an annual address
* a card showing how this president’s addresses rank in terms of vocabulary density, length, giving examples of unique words or recurring themes, etc.
* a card showing a particular quote from this president’s addresses
* a card displaying general information about this president’s tenure in office, perhaps including contemporary news events, how they delivered the addresses (in person or written), and other interesting facts or figures about their presidency
* a timeline showing this president and those immediately proceeding and succeeding him

Users may navigate through the info cards, access the main menu, or return to the main screen by clicking the SOTU-db logo in the top banner.

*Note: in some cases, a name may be ambiguous; in these cases a disambiguation or special comparison page could be used in response to particular names. For example, typing “Roosevelt” could provide a card with links to either Theodore or Franklin Roosevelt, and an additional card* *showing relevant comparisons between the two.*

### YEAR landing page

*access by typing 4-digit year into search box*

This page would automatically bring a user directly to one particular object in the collection: the speech from the year entered.

*Note: in some cases, one year may have multiple addresses. In these cases, a special card could be brought up comparing the two speeches.*

### TEXT landing page

*access by typing any term not a name or year into search box*

The text landing page is expected to be the primary means of interaction with the software for many users. The text landing page will feature a series of cards containing, for example:

* a visual graph of the usage of this word over the history of SOTU
* a card featuring the total number of times this term appears in the corpus, and a rank of how popular this word is within the corpus
* cards indicating which president said this term first, most often, most recently, etc.
* a “context” card giving example(s) of how this term has been used in the context of surrounding words in the corpus
* a card showing topics/words that closely relate to this words across the corpus

Each info card would be clickable to drill down and explore the content further.

### Info Cards

*access via main screen, under search bar*

The info cards on the main screen are designed to engage users, encourage playful interactivity, and guide users who may be unfamiliar with the platform or come without a specific research question in mind. These info cards could consist of some of the following categories:

* **quote cards** that show a provocative or inspiring quote. Users would click to find out who said the quote, leading them to a text landing page with the quote pre-filled.
* **Twitter cards** that pull in tweets featuring the SOTU-db handle or hashtag
* **did you know? cards** that contain an amusing or interesting piece of trivia about a particular president or address
* **guess who cards** that feature a statistic (e.g. longest address, used a unique word, etc) and ask the user to click to find out which president or speech the statistic applies to
* **ranking** cards that show the top 3-5 most popular, longest, shortest, most cited, etc. speeches
* **visualization** cards that show a visualization and ask the user to guess which word it applies to, or some other form of engagement

### Main menu

*access via “hamburger menu” anywhere in app/on site*

The Main Menu will include:

* Main logo / link back to main screen
* “About” button with link to about screen
* User login management and history\*
* Development credit, license information, and URLs

### About screen

*access via “about” link on main screen, or “About” menu item*

The About screen will include a general tutorial text and video explanation outlining the basic functionality of the site. It will also include a link to further online documentation / help, if available.

## Performance Requirements

This project has no target for concurrent user support or operating capacity. These variables will be determined by the resources at the project’s disposal and learning more about these capacities will make up a valuable component of this project.

## Database Requirements

This project will make use of a corpus of texts (.txt files) that will need to be stored, accessed, and manipulated by the software. Also, users will generate search queries that will need to be stored and processed by a database. I’m unsure if I’m adding user login / search history functionality at this time.

## Design constraints

This section outlines constraints imposed by other standards, hardware limitations, etc.

### Standards compliance

This software is not required to nor designed to interact or interface with other software platforms or standards beyond its accessibility through Chrome web browser and Android 7.1.1 mobile operating system.

### Software system attributes

#### Reliability

Software reliability will be determined by resources at the project’s disposal, which are undetermined at this point.

#### Availability

The software’s availability will be determined by resources at the project’s disposal, which are undetermined at this point. Ideally, the system should be available for use 24/7 with 100% uptime.

#### Security

Security requirements relate directly to the use of user login information. If users interact anonymously with the platform, little is needed in terms of security. If users are able to login and save search history or other information, site security will need to be revisited.

#### Maintainability

The design of this project relies upon various components/modules, many of which have functionality that is duplicated by other pieces of software. Therefore, the development approach of this project will be to create a functional model based primarily on existing resources, then “drop-in” project-created components/modules as they are created. Because of this approach, maintainability of the project should be quite good, as components can be “dropped out” if maintaining them becomes difficult. There is no guarantee, however, that the underlying or preexisting tools will continue to function indefinitely.

#### Portability

The portability of this program will be determined by the language, tools, hosting platform, etc which are chosen to run the software.

# Appendices

## Appendix X: References