ZipCompare

Software Requirements Specification (SRS)

[1 Scope 3](#_1v1yuxt)

[1.1 Identification 3](#_4f1mdlm)

[1.2 System overview 3](#_2u6wntf)

[1.3 Document overview 3](#_19c6y18)

[2 Referenced documents 3](#_3tbugp1)

[3 CSCI-wide design decisions 3](#_28h4qwu)

[4 CSCI architectural design 4](#_nmf14n)

[4.1 CSCI components 4](#_37m2jsg)

[4.2 Concept of execution 5](#_1mrcu09)

[4.3 Interface design 5](#_46r0co2)

[4.3.1 Interface identification and diagrams 6](#_2lwamvv)

[4.3.2 (Project-unique identifier of interface) 6](#_111kx3o)

[5 CSCI detailed design 7](#_3l18frh)

[5.1 (Project-unique identifier of a software unit, or designator of a group of software units) 8](#_206ipza)

[6 Requirements traceability 9](#_4k668n3)

[7 Notes 9](#_1fob9te)

[A. Appendixes 10](#_3fwokq0)

# Scope

This section shall be divided into the following paragraphs.

## Identification

This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

This system is the product resulting from the University of Maryland, Baltimore County (UMBC) Computer Science 447: Software Engineering I course, to produce a predetermined product for an assigned customer. The current working title is ZipCompare. The version is 1.0 as defined by the current progress.

## System overview

This paragraph shall briefly state the purpose of the system and the software to which this document applies. It shall describe the general nature of the system and software; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

This software application is designed to identify certain details of a zip code that is requested by the user. After being assigned this project by Professor Russell Cain, the group of developers were placed together based on qualities determined through survey.

The system is sponsored by the course Computer Science 447: Software Engineering of University of Maryland, Baltimore County. The customer of the product is Michael Neary in the College of Engineering and Information Technology. The developers are Keith McNamara Jr., Dustin Cuocci, Benjamin Decre, William Delas Penas, DJ Streat, and Mitchell Borrego. All code was produced and tested on UMBC campus. The acquirers are Professor Russell Cain, as the instructor for the course, and our customer, Michael Neary.

## Document overview

This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

This document’s, Software Requirements Specification, purpose is to describe to the appropriate audience the requirements given to our team from the customer. The document also briefly explains the methodology our team will use to satisfy the requirements given from our customer.

# Referenced documents

This section shall list the number, title, revision, and date of all documents referenced in this specification. This section shall also identify the source for all documents not available through normal Government stocking activities.

# Requirements

This section shall be divided into the following paragraphs to specify the CSCI requirements, that is, those characteristics of the CSCI that are conditions for its acceptance. CSCI requirements are software requirements generated to satisfy the system requirements allocated to this CSCI. Each requirement shall be assigned a project-unique identifier to support testing and traceability and shall be stated in such a way that an objective test can be defined for it. Each requirement shall be annotated with associated qualification method(s) (see section [4)](#_41mghml) and traceability to system (or subsystem, if applicable) requirements (see section [5.a)](#_vx1227) if not provided in those sections. The degree of detail to be provided shall be guided by the following rule: Include those characteristics of the CSCI that are conditions for CSCI acceptance; defer to design descriptions those characteristics that the acquirer is willing to leave up to the developer. If there are no requirements in a given paragraph, the paragraph shall so state. If a given requirement fits into more than one paragraph, it may be stated once and referenced from the other paragraphs.

## Required states and modes

If the CSCI is required to operate in more than one state or mode having requirements distinct from other states or modes, this paragraph shall identify and define each state and mode. Examples of states and modes include: idle, ready, active, post- use analysis, training, degraded, emergency, backup, wartime, peacetime. The distinction between states and modes is arbitrary. A CSCI may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If no states or modes are required, this paragraph shall so state, without the need to create artificial distinctions. If states and/or modes are required, each requirement or group of requirements in this specification shall be correlated to the states and modes. The correlation may be indicated by a table or other method in this paragraph, in an appendix referenced from this paragraph, or by annotation of the requirements in the paragraphs where they appear.

## CSCI capability requirements

This paragraph shall be divided into subparagraphs to itemize the requirements associated with each capability of the CSCI. A "capability" is defined as a group of related requirements. The word "capability" may be replaced with "function," "subject," "object," or other term useful for presenting the requirements.

### (CSCI capability)

This paragraph shall identify a required CSCI capability and shall itemize the requirements associated with the capability. If the capability can be more clearly specified by dividing it into constituent capabilities, the constituent capabilities shall be specified in subparagraphs. The requirements shall specify required behavior of the CSCI and shall include applicable parameters, such as response times, throughput times, other timing constraints, sequencing, accuracy, capacities (how much/how many), priorities, continuous operation require- ments, and allowable deviations based on operating conditions. The requirements shall include, as applicable, required behavior under unexpected, unallowed, or "out of bounds" conditions, requirements for error handling, and any provisions to be incorporated into the CSCI to provide continuity of operations in the event of emergencies. Paragraph [3.3.x](#_lnxbz9) of this DID provides a list of topics to be considered when specifying requirements regarding inputs the CSCI must accept and outputs it must produce.

The system shall be a point-and-click functioning application.

* The system shall pick out a zip code based on the area selected by the user. The application shall display key information of the desired zip code.
* The system shall operate on external mouse functionality to support the point-and-click requirement. The system shall take in a click input on the map provided, and zoom into the specific zip code to display information about that zipcode.
* The system shall notify the user if an invalid zip code has been selected by the user.

The system shall have search functionality for the user to be able to type in a specific zip code to find it on the map provided.

* The system shall display key information from the desired zip code.
* The system shall notify a user if an invalid zip code has been entered by the user.

The system shall support databases to maintain user accounts.

* The system shall maintain a databases to organize users and their account-specific details. The databases shall hold information such as the usernames and passwords, their respective zip codes, and their previous searches.

The system shall maintain/support a comparison system.

* The system shall compare, upon user selection, up to three different zip codes. The comparison feature shall show the key differences among similar attributes of the zip codes. The zip codes with more positive ~~a~~ features will earn a higher score. The zip code that has the higher overall score will be ranked at the top from the result of the comparison.
* The system shall support an upvote feature. The system shall handle the upvote feature by only allowing users to upvote their current zip code. This shall ensure no user and falsely upvote a zip code of which they have no experience with.
* The system shall display the top five upvoted zip codes chosen by the entire user base.

The system shall maintain a database that stores our data on the zipcodes.

* The system will use external resources to gather the data on the zip codes.
* The system shall update its data after a certain period of time (not determined).

## CSCI external interface requirements

This paragraph shall be divided into subparagraphs to specify the requirements, if any, for the CSCI’s external interfaces. This paragraph may reference one or more Interface Requirements Specifications (IRSs) or other documents containing these requirements.

### Interface identification and diagrams

This paragraph shall identify the required external interfaces of the CSCI (that is, relationships with other entities that involve sharing, providing or exchanging data). The identification of each interface shall include a project-unique identifier and shall designate the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification shall state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams shall be provided to depict the interfaces.

The application shall be a web-based system. It shall utilize mouse point-and-click functionality.

### (Project-unique identifier of interface)

This paragraph (beginning with 3.3.2) shall identify a CSCI external interface by project-unique identifier, shall briefly identify the interfacing entities, and shall be divided into subparagraphs as needed to state the requirements imposed on the CSCI to achieve the interface. Interface characteristics of the other entities involved in the interface shall be stated as assumptions or as "When [the entity not covered] does this, the CSCI shall...," not as requirements on the other entities. This paragraph may reference other documents (such as data dictionaries, standards for communication protocols, and standards for user interfaces) in place of stating the information here. The requirements shall include the following, as applicable, presented in any order suited to the requirements, and shall note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):

1. Priority that the CSCI must assign the interface
2. Requirements on the type of interface (such as real-time data transfer, storage-and- retrieval of data, etc.) to be implemented
3. Required characteristics of individual data elements that the CSCI must provide, store, send, access, receive, etc., such as:
   1. Names/identifiers
      1. Project-unique identifier
      2. Non-technical (natural-language) name
      3. DoD standard data element name
      4. Technical name (e.g., variable or field name in code or database)
      5. Abbreviation or synonymous names
   2. Data type (alphanumeric, integer, etc.)
   3. Size and format (such as length and punctuation of a character string)
   4. Units of measurement (such as meters, dollars, nanoseconds)
   5. Range or enumeration of possible values (such as 0-99)
   6. Accuracy (how correct) and precision (number of significant digits)
   7. Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply
   8. Security and privacy constraints
   9. Sources (setting/sending entities) and recipients (using/receiving entities)
4. Required characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the CSCI must provide, store, send, access, receive, etc., such as:
5. Names/identifiers
   * 1. Project-unique identifier
     2. Non-technical (natural language) name
     3. Technical name (e.g., record or data structure name in code or database)
     4. Abbreviations or synonymous names
6. Data elements in the assembly and their structure (number, order, grouping)
7. Medium (such as disk) and structure of data elements/assemblies on the medium
8. Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)
9. Relationships among assemblies, such as sorting/access characteristics
10. Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply
11. Security and privacy constraints
12. Sources (setting/sending entities) and recipients (using/receiving entities)
13. Required characteristics of communication methods that the CSCI must use for the interface, such as:
    1. Project-unique identifier(s)
    2. Communication links/bands/frequencies/media and their characteristics
    3. Message formatting
    4. Flow control (such as sequence numbering and buffer allocation)
    5. Data transfer rate, whether periodic/aperiodic, and interval between transfers
    6. Routing, addressing, and naming conventions
    7. Transmission services, including priority and grade
    8. Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing
14. Required characteristics of protocols the CSCI must use for the interface, such as:
    1. Project-unique identifier(s)
    2. Priority/layer of the protocol
    3. Packeting, including fragmentation and reassembly, routing, and addressing
    4. Legality checks, error control, and recovery procedures
    5. Synchronization, including connection establishment, maintenance, termination
    6. Status, identification, and any other reporting features
15. Other required characteristics, such as physical compatibility of the interfacing entities (dimensions, tolerances, loads, plug compatibility, etc.), voltages, etc.

## CSCI internal interface requirements

This paragraph shall specify the requirements, if any, imposed on interfaces internal to the CSCI. If all internal interfaces are left to the design, this fact shall be so stated. If such requirements are to be imposed, paragraph [3.3](#_17dp8vu) of this DID provides a list of topics to be considered.

## CSCI internal data requirements

This paragraph shall specify the requirements, if any, imposed on data internal to the CSCI. Included shall be requirements, if any, on databases and data files to be included in the CSCI. If all decisions about internal data are left to the design, this fact shall be so stated. If such requirements are to be imposed, paragraphs [3.3.x.c](#_35nkun2) and [3.3.x.d](#_1ksv4uv) of this DID provide a list of topics to be considered.

A user attempting to login, must already have a valid username and password to use the application. Should the user not already have an account, he or she must create using the tools provided by the system.

The user must enter a valid zip code to receive information displayed on the screen.

A user cannot upvote a zip code that they do not currently occupy. This is to prevent false positive style upvotes from user that have no affiliation with a certain area.

## Adaptation requirements

This paragraph shall specify the requirements, if any, concerning installation-dependent data to be provided by the CSCI (such as site-dependent latitude and longitude or site-dependent state tax codes) and operational parameters that the CSCI is required to use that may vary according to operational needs (such as parameters indicating operation-dependent targeting constants or data recording).

Users should have up-to-date versions of web browsers (e.g. Google Chrome, Mozilla Firefox, Safari)

The system relies on external resources to determine the statistics of the zip codes. The system will do its best to maintain updated statistics to provide users the most accurate data.

## Safety requirements

This paragraph shall specify the CSCI requirements, if any, concerned with preventing or minimizing unintended hazards to personnel, property, and the physical environment. Examples include safeguards the CSCI must provide to prevent inadvertent actions (such as accidentally issuing an "auto pilot off" command) and non-actions (such as failure to issue an intended "auto pilot off" command). This paragraph shall include the CSCI requirements, if any, regarding nuclear components of the system, including, as applicable, prevention of inadvertent detonation and compliance with nuclear safety rules.

## Security and privacy requirements

This paragraph shall specify the CSCI requirements, if any, concerned with maintaining security and privacy. These requirements shall include, as applicable, the security/privacy environment in which the CSCI must operate, the type and degree of security or privacy to be provided, the security/privacy risks the CSCI must withstand, required safeguards to reduce those risks, the security/privacy policy that must be met, the security/privacy accountability the CSCI must provide, and the criteria that must be met for security/privacy certification/accreditation.

## CSCI environment requirements

This paragraph shall specify the requirements, if any, regarding the environment in which the CSCI must operate. Examples include the computer hardware and operating system on which the CSCI must run. (Additional requirements concerning computer resources are given in the next paragraph.)

## Computer resource requirements

This paragraph shall be divided into the following subparagraphs.

### Computer hardware requirements

This paragraph shall specify the requirements, if any, regarding computer hardware that must be used by the CSCI. The requirements shall include, as applicable, number of each type of equipment, type, size, capacity, and other required characteristics of processors, memory, input/output devices, auxiliary storage, communications/network equipment, and other required equipment.

### Computer hardware resource utilization requirements

This paragraph shall specify the requirements, if any, on the CSCI’s computer hardware resource utilization, such as maximum allowable use of processor capacity, memory capacity, input/output device capacity, auxiliary storage device capacity, and communications/network equipment capacity. The requirements (stated, for example, as percentages of the capacity of each computer hardware resource) shall include the conditions, if any, under which the resource utilization is to be measured.

### Computer software requirements

This paragraph shall specify the requirements, if any, regarding computer software that must be used by, or incorporated into, the CSCI. Examples include operating systems, database management systems, communications/ network software, utility software, input and equipment simulators, test software, and manufacturing software. The correct nomenclature, version, and documentation references of each such software item shall be provided.

The system shall be compatible with most modern operating systems, mainly including Windows, Mac OS, and Linux. The databases being used shall be managed through MongoDB.

### Computer communications requirements

This paragraph shall specify the additional requirements, if any, concerning the computer communications that must be used by the CSCI. Examples include geographic locations to be linked; configuration and network topology; transmission techniques; data transfer rates; gateways; required system use times; type and volume of data to be transmitted/received; time boundaries for transmission/reception/response; peak volumes of data; and diagnostic features.

The system shall be connected to an API, however internet access is required for the system to connect to the proper servers. The databases must be accessed to connect to the user’s account. A proper internet connection is also necessary to open the application in a web-browser setting.

## Software quality factors

This paragraph shall specify the CSCI requirements, if any, concerned with software quality factors identified in the contract or derived from a higher level specification. Examples include quantitative requirements regarding CSCI functionality (the ability to perform all required functions), reliability (the ability to perform with correct, consistent results), maintainability (the ability to be easily corrected), availability (the ability to be accessed and operated when needed), flexibility (the ability to be easily adapted to changing requirements), portability (the ability to be easily modified for a new environment), reusability (the ability to be used in multiple applications), testability (the ability to be easily and thoroughly tested), usability (the ability to be easily learned and used), and other attributes.

The system is required to be able to produce a map of the United States.

The system is required to be able to respond to mouse clicks on the maps.

## Design and implementation constraints

This paragraph shall specify the requirements, if any, that constrain the design and implementation of the CSCI. These requirements may be specified by reference to appropriate commercial or military standards and specifications. Examples include requirements concerning:

1. Use of a particular CSCI architecture or requirements on the architecture, such as required databases or other software units; use of standard, military, or existing components; or use of Government/acquirer-furnished property (equipment, information, or software) The system’s map features and design will be limited by Google’s API.
2. Use of particular design or implementation standards; use of particular data standards; use of a particular programming language. The developers plan to use Python as the primary programming language.
3. Flexibility and expandability that must be provided to support anticipated areas of growth or changes in technology, threat, or mission

## Personnel-related requirements

This paragraph shall specify the CSCI requirements, if any, included to accommodate the number, skill levels, duty cycles, training needs, or other information about the personnel who will use or support the CSCI. Examples include requirements for number of simultaneous users and for built-in help or training features. Also included shall be the human factors engineering requirements, if any, imposed on the CSCI. These requirements shall include, as applicable, considerations for the capabilities and limitations of humans; foreseeable human errors under both normal and extreme conditions; and specific areas where the effects of human error would be particularly serious. Examples include requirements for color and duration of error messages, physical placement of critical indicators or keys, and use of auditory signals.

The system is required to be applicable to users with a wide range of technical knowledge. Thus, the system’s functionality is simplified to allow users with little to no technical language to use the system. Most of the user’s participation in the system are clicking, cursor movement, and typing information into the system. The information refers to the user’s username, password, and the zip codes they want to view.

## Training-related requirements

This paragraph shall specify the CSCI requirements, if any, pertaining to training. Examples include training software to be included in the CSCI.

## Logistics-related requirements

This paragraph shall specify the CSCI requirements, if any, concerned with logistics considerations. These considerations may include: system maintenance, software support, system transportation modes, supply-system requirements, impact on existing facilities, and impact on existing equipment.

## Other requirements

This paragraph shall specify additional CSCI requirements, if any, not covered in the previous paragraphs.

## Packaging requirements

This section shall specify the requirements, if any, for packaging, labeling, and handling the CSCI for delivery (for example, delivery on 8 track magnetic tape labelled and packaged in a certain way). Applicable military specifications and standards may be referenced if appropriate.

## Precedence and criticality of requirements

This paragraph shall specify, if applicable, the order of precedence, criticality, or assigned weights indicating the relative importance of the requirements in this specification. Examples include identifying those requirements deemed critical to safety, to security, or to privacy for purposes of singling them out for special treatment. If all requirements have equal weight, this paragraph shall so state.

# Qualification provisions

This section shall define a set of qualification methods and shall specify for each requirement in Section [3](#_tyjcwt) the method(s) to be used to ensure that the requirement has been met. A table may be used to present this information, or each requirement in Section [3](#_tyjcwt) may be annotated with the method(s) to be used. Qualification methods may include:

1. Demonstration: The operation of the CSCI, or a part of the CSCI, that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.
2. Test: The operation of the CSCI, or a part of the CSCI, using instrumentation or other special test equipment to collect data for later analysis.
3. Analysis: The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpretation, or extrapolation of test results.
4. Inspection: The visual examination of CSCI code, documentation, etc.
5. Special qualification methods: Any special qualification methods for the CSCI, such as special tools, techniques, procedures, facilities, and acceptance limits.

# Requirements traceability

This paragraph shall contain:

1. Traceability from each CSCI requirement in this specification to the system (or subsystem, if applicable) requirements it addresses. (Alternatively, this traceability may be provided by annotating each requirement in Section [3.)](#_tyjcwt)

Note: Each level of system refinement may result in requirements not directly traceable to higher-level requirements. For example, a system architectural design that creates multiple CSCIs may result in requirements about how the CSCIs will interface, even though these interfaces are not covered in system requirements. Such requirements may be traced to a general requirement such as "system implementation" or to the system design decisions that resulted in their generation.

1. Traceability from each system (or subsystem, if applicable) requirement allocated to this CSCI to the CSCI requirements that address it. All system (subsystem) requirements allocated to this CSCI shall be accounted for. Those that trace to CSCI requirements contained in Interface Requirements Specifications (IRSs) shall reference those IRSs.

# Notes

This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

# Appendixes

Appendixes may be used to provide information published separately for convenience in document maintenance (e.g., charts, classified data). As applicable, each appendix shall be referenced in the main body of the document where the data would normally have been provided. Appendixes may be bound as separate documents for ease in handling. Appendixes shall be lettered alphabetically (A, B, etc.).

DESCRIPTION/PURPOSE

The Software Design Description (SDD) describes the design of a Computer Software Configuration Item (CSCI). It describes the CSCI-wide design decisions, the CSCI architectural design, and the detailed design needed to implement the software. The SDD may be supplemented by Interface Design Descriptions (IDDs) and Database Design Descriptions (DBDDs).

APPLICATION/INTERRELATIONSHIP

Portions of this plan may be bound separately if this approach enhances their usability. Examples include plans for software configuration management and software quality assurance.

The Contract Data Requirements List (CDRL) should specify whether deliverable data are to be delivered on paper or electronic media; are to be in a given electronic form (such as ASCII, CALS, or compatible with a specified word processor or other support software); may be delivered in developer format rather than in the format specified herein; and may reside in a computer-aided software engineering (CASE) or other automated tool rather than in the form of a traditional document.

PREPARATION INSTRUCTIONS

General instructions.

a. Automated techniques. Use of automated techniques is encouraged. The term "document" in this means a collection of data regardless of its medium.

b. Alternate presentation styles. Diagrams, tables, matrices, and other presentation styles are acceptable substitutes for text when data required can be made more readable using these styles.

c. Title page or identifier. The document shall include a title page containing, as applicable: document number; volume number; version/revision indicator; security markings or other restrictions on the handling of the document; date; document title; name, abbreviation, and any other identifier for the system, subsystem, or item to which the document applies; contract number; CDRL item number; organization for which the document has been prepared; name and address of the preparing organization; and distribution statement. For data in a database or other alternative form, this information shall be included on external and internal labels or by equivalent identification methods.

d. Table of contents. The document shall contain a table of contents providing the number, title, and page number of each titled paragraph, figure, table, and appendix. For data in a database or other alternative form, this information shall consist of an internal or external table of contents containing pointers to, or instructions for accessing, each paragraph, figure, table, and appendix or their equivalents.

e. Page numbering/labeling. Each page shall contain a unique page number and display the document number, including version, volume, and date, as applicable. For data in a database or other alternative form, files, screens, or other entities shall be assigned names or numbers in such a way that desired data can be indexed and accessed.

f. Response to tailoring instructions. If a paragraph is tailored out of this document, the resulting document shall contain the corresponding paragraph number and title, followed by "This paragraph has been tailored out." For data in a database or other alternative form, this representation need occur only in the table of contents or equivalent.

g. Multiple paragraphs and subparagraphs. Any section, paragraph, or subparagraph in this DID may be written as multiple paragraphs or subparagraphs to enhance readability.

h. Standard data descriptions. If a data description required by this document has been published in a standard data element dictionary specified in the contract, reference to an entry in that dictionary is preferred over including the description itself.

i. Substitution of existing documents. Commercial or other existing documents, including other project plans, may be substituted for all or part of the document if they contain the required data.