**CPH-IVT High Level Requirements v0.7**

Authors: Shayne Parker, Phil Pfeiffer, Ethan Spangler, and Ed Wilson

Last Updated 13 November 2020

Contents

[1 Purpose 1](#_Toc56136143)

[2 Requirements 1](#_Toc56136144)

[2.1 Functionality 1](#_Toc56136145)

[2.2 Role-Based Access to Functionality 3](#_Toc56136146)

[2.2.1 Role-based access 3](#_Toc56136147)

[2.2.2 Access by role 4](#_Toc56136148)

[2.3 User interface 4](#_Toc56136149)

[2.4 Logging 4](#_Toc56136150)

[2.5 User Support 4](#_Toc56136151)

[3 Limitations 4](#_Toc56136152)

[4 Uncertainties 5](#_Toc56136153)

# Purpose

The proposed system, hereafter referred to as **CPH-IVT**, shall visualize health data information and trends for discrete geographical region, including user-defined regions. Requirements that are outside of the scope of the system’s first iteration are highlighted in yellow.

# Requirements

## Functionality

1. CPH-IVT shall enable the transparent **retrieval and storage of health indicator data** from the [County Health Rankings](https://www.countyhealthrankings.org/) into a persistent data store
   1. This data store shall be part of the CPH-IVT system
   2. The system shall store data for all U.S. counties, by year
   3. CPH-IVT shall enable this capability to be automatic or manual
2. CPH-IVT shall enable the use of the following queries to **enumerate the database’s counties**
   1. CPH-IVT shall support the enumeration of counties by any combination of the following
      1. FIPS code
      2. Name, by regular expression
      3. State name – by set, or “any”
   2. The query shall return all matching counties, including each county’s FIPS code, name, and state
3. CPH-IVT shall enable the use of the following queries to **enumerate the database’s health indicators**
   1. CPH-IVT shall support the enumeration of counties by any combination of the following
      1. Health indicator code, by regular expression
      2. description, by regular expression
      3. year, by set or range of years
   2. The query shall return all matching counties, including each indicator’s indicator code, description, and years
4. CPH-IVT shall enable the **management of health data by** user-specified groupings of counties (**regions**).
   1. CPH-IVT shall support three types of regions
      1. Globally available regions, including
         1. Regions that are generated automatically from CPH-IVT data
            1. These regions shall include counties by state
         2. Regions that are custom-created for global distribution
      2. Locally available regions, custom-created for local use
         1. These regions shall only be visible from the sites at which they’re created
   2. CPH-IVT shall associate each region with three attributes
      1. A unique identifier (**UID**)
      2. A name for that region
      3. A description of that region’s content
   3. CPH-IVT shall enable the use of operations to access and manipulate regions
      1. The following operations shall be supported, as transactions
         1. Region-enumerating transactions (i.e., queries)
            1. CPH-IVT shall support the enumeration of regions by any combination of the following

Scope - auto-generated (i.e., states) or custom

County, by FIPS codes, with up to three qualifiers

Must include all of

* Note: specifying this without the “any of “ parameter yields an “any” query

Must include at least one of

* Note: specifying this without the “all of “ parameter yields an “any” query

Must not include any of

Name, by regular expression

Description, by regular expression

Set of region UIDs

* + - * 1. Output: in each case, the matching regions, if any, each with its UID, scope, name, description, and counties, each of which is listed with its name, state, and FIPS code
      1. Region-forming operations
         1. CPH-IVT should support the creation of regions by any combination of the following

Region union – i.e., union of multiple available regions and county FIPS codes

Region intersection – i.e., intersection of multiple available regions and county FIPS codes

Region difference – i.e., counties in a first region that are not in a second

* Note: treat removing a list of counties from a region as a special case of region difference
  + - * 1. Parameters

the current regions’ UIDs

a name for the new region

a description of the new region’s content

* + - * 1. The output shall be a region with a set of associated counties – i.e., regions shall not nest
      1. CPH-IVT shall support the updating of regions by any combination of the following
         1. The operations per se

Region union – i.e., union of multiple available regions to update a target region

Region intersection – i.e., intersection of multiple available regions to update a target region

Region difference – i.e., counties in a first region that are not in a second

* Note: treat removing a list of counties from a region as a special case of region difference
  + - * 1. Parameters

the UID for the region to modify

the UIDs for the supporting regions

(optional) a revised name for the new region

(optional) a revised description of the new region’s content

* + - 1. Region deletion operations
         1. CPH-IVT should support the deletion of regions by any combination of the following

Scope - custom only, except for corruption

County, by FIPS code, except for corruption

Must include all of

* Note: specifying this without the “any of “ parameter yields an “any” query

Must include at least one of

* Note: specifying this without the “all of “ parameter yields an “any” query

Must not include any of

Name, by regular expression

Description, by regular expression

Set of UIDs

* + - * 1. Output: in each case, the deleted regions, if any, each with its UID scope, name, description, and counties, each of which is listed with its name, state, and FIPS code
* Note: since some deletions are dangerous, special care may be needed to implement them

1. CPH-IVT shall support the use of **queries to retrieve individual indicators**
   1. CPH-IVT shall support the retrieval of health data by any of the following attributes
      1. A set of health indicator UIDs, possibly including regular expressions
      2. A set of years, possibly including ranges
      3. A set of locales, including
         1. A (possibly empty) set of regions by region name or region UID
         2. A (possibly empty) set of counties by (county, state) name or FIPS code
      4. Description, by regular expression
      5. By items in client-supplied list
   2. The query shall return a set of quadruples, consisting of

For every year in the set of years

For every indicator in the set of indicator UIDs

For every county in every locale:

A (year, county FIPS code, indicator UID, indicator value) quadruple

* 1. The query shall return a set of pairs, consisting of

For a specified year (default: current)

For every indicator in the set of indicator UIDs (typically, one; two for scatter plots)

For every county in every locale:

A (county FIPS code, indicator UID, indicator value) triple

* Note: this will be pairs if only one indicator is supplied

1. CPH-IVT shall support the use of **graphs** to present this data
   1. The system shall support line graphs, with X-axis as percentile and Y-axis as value, with counties as points
   2. The system shall support scatter plots (second iteration), with the two indicators as X and Y axes

## Role-Based Access to Functionality

### Role-based access

1. CPH-IVT shall support **role-based access** to its functionality and content
   1. Two classes of roles shall be supported
      1. A default role for public users
      2. Two roles for registered users
         1. System administrators
         2. Data administrators
   2. Duly authorized registered users shall be allowed to assume multiple roles
      1. Users shall assume only one role at a time and be limited to those actions permitted for that role
2. CPH-IVT shall **manage information on registered users**
   1. CPH-IVT shall store the following information for registered users
      1. ETSU E-number
      2. CPH-IVT registered roles
   2. CPH-IVT will support local authorization for privileged users
   3. CPH-IVT shall use ETSU’s active directory to access all other information pertaining to registered users
3. **Access** to CPH-IVT for registered users **shall be authorized** on a per-session basis
   1. CPH-IVT shall authenticate all privileged users before affording them access to the system
      1. ETSU’s authentication process shall be used to authenticate users
      2. CPH-IVT shall authorize privileged users locally
      3. The authentication process shall include associating each user with a role type
   2. CPH-IVT shall support auto-logoff for users whose sessions are idle for a client-determined period of time

### Access by role

1. CPH-IVT shall enable system administrators to access and manipulate all system persistent content, including system logs

* ***Need to flesh this out more carefully***

1. CPH-IVT shall enable data administrators to access and manipulate all CPH data and global and local regions

* ***Need to flesh this out more carefully***

1. CPH-IVT shall enable public users to access all CPH data, all CPH global regions, and to access and manipulate local regions

* ***Need to flesh this out more carefully***

## User interface

1. CPH-IVT’s user interface shall **comply with ADA guidelines** for system accessibility
2. CPH-IVT’s user interface shall display a statement **disclaiming responsibility** for the quality of the data it presents and any interfaces that users draw from that data

## Logging

1. CPH-IVT shall support the discretionary **logging of the system’s operations**
   1. This log shall be maintained within the CPH-IVT data store

* ***Need to think about what the log can capture and how it can be set to capture different levels of operations***

1. CPH-IVT shall gather metrics on system use

## User Support

1. CPH-IVT shall be delivered with an automated installation setup script.
   1. Data fetching shall be done at time of installation.
   2. Data upload shall be done immediately after fetching.
   3. The CPH-IVT application shall be launched immediately after data upload.
2. CPH-IVT shall be delivered with **user documentation**.
   1. “How to Use” for public users
   2. “How to Administrate” for administrators
   3. Technical documentation for IT staff.

# Limitations

The following issues shall be treated as outside the scope of CPH-IVT’s requirements:

1. CPH-IVT takes no responsibility for its presentation of CHR data, including the data proper.
   1. CPH-IVT will do no data cleansing, data validation, or data analysis
2. CPH-IVT takes no responsibility for changes to CHR counties over time.
   1. States may add, modify, or abolish counties. CPH-IVT assumes the user is the domain expert and more knowledgeable about what counties exist and their location than the CPH-IVT developers.
      1. Following the above assumption, CPH-IVT leaves it to the user to select the appropriate counties and regions.
3. CPH-IVT takes no responsibility for changes to CHR regions over time.
   1. Definitions of regions may shift over time, including regions like, say, southwest Virginia, that have been used to define larger regions, like, say, central Appalachia. CPH-IVT assumes the user is the domain expert and more knowledgeable about what regions exist and their location than the CPH-IVT developers.
      1. Following the above assumption, CPH-IVT leaves it to the user to keep county membership in regions current.
4. CPH-IVT takes no responsibility for changes to CHR health indicators over time.
   1. CHR may change the descriptions of health indicators between years. CPH-IVT does not attempt to trace these changes between years.
      1. All data will still be presented, but CPH-IVT leaves it to the user to select appropriate indicators to view.
5. CPH-IVT should support the use of a write-once, read-many log. However, implementing such a log is beyond the current project’s scope, due to lack of resources.
6. CPH-IVT should support a dual-database architecture, where a relational database maintains transient data, like accounts, and a NoSQL database maintains CHR data. Implementing this architecture, however, is also beyond the current project’s scope, due to lack of resources.

# Uncertainties

The following issues need clarification from the client before classifying as in-scope or out-of-scope:

1. What temporal information should be associated with regions, if any?

* Counties have changed names and status over time.
* Due to these changes, a built-in region for, say, Virginia or South Dakota will name different counties, depending on the year in which a region is defined
* Given these changes, what should the system do for the two types of regions: i.e.,
* For automatically generated regions, like states
* Should the system track each region’s current counties, and require users to hand-tailor references to sets of counties for previous years
* Should the system associate each region with a year, and interpret queries relative to years?
* For custom-created regions
* Should the system treat each region as current and flag uses of regions that name counties that are no longer current?
* Should the system allow users to associate years with the regions they create, defaulting to the current year?

1. How should locally available regions be accessible?

* Should they be specific to a platform – i.e., accessible from any browser that any user of a given platform invokes?
* Should they be specific to a platform’s user—i.e., accessible from any browser that a given user of a given platform invokes?
* Should they be specific to a specific combination of user, browser, and platform?