IMPLEMENTATION GUIDE FOR THE REQMODEL SYSTEM

Electron-Ion Collider Project

1. OVERVIEW

The ReqModel system is an application developed at Jefferson Lab that converts a collection of properly formatted Excel spreadsheet into a system of webpages containing requirement and interface information. The system was created to make requirement information, which is collected in a restricted system, available to all scientists, engineers, and contributors.

The process described in this document contains the steps necessary to ensure that the webpages are updated on a regular basis and contain the most recent data. The process consists of three major steps: source insertion, source processing, and document deployment. Each of these steps are described in the sections below. A quick reference guide to servers, directories, and files where data is stored is listed in table 1.

Table 1. Location and Description of Key Data

Name	Location	Description
ReqModel Application	infra:/var/ReqModel/bin	This is the binary application, configuration file, and cron script that is used to interpret the spreadsheets and generate the web pages.
ReqModel Source	infra:/var/ReqModel/src	The source code for the currently deployed version of the ReqModel system.
Data Sources	/sgroup/phys_div/Transfer/EIC-Requirements	This is the directory where the requirements spreadsheets (in CSV format) that are ready for processing will be stored.
Data Archives	/sgroup/phys_div/Transfer/EIC-Requirements/archive	Once the spreadsheets have been processed, this is the directory where they will be copied and retained for 1 year.
HTML Output Path	/sgroup/phys_div/www/eic/html/Requirements	This is the root path for the directory that will contain all of the requirement web pages, scripts, css files and supporting data.
Web Site	https://eic.jlab.org/Requirements	This is the base page for the web pages that are produced by the ReqModel system. References to this page are typically provided at https://eic.jlab.org
Source CSV	https://eic.jlab.org/Requirements/source.csv	This is a copy of the CSV file that was used to generate the requirements page. It will be installed here for use by other applications that required access to the current data.

2. SOURCE INSERTION

The ReqModel process is triggered when the CSV data files are installed in the data sources directory (/sgroup/phys_div/Transfer/EIC-Requirements). The data sources consist of 3 files: the requirements file, the links file, and the sub-systems file. The naming conventions and required format for these files is described below.

2.1 requirements.csv

This file contains the current requirements. The file must meet the following criteria.

a. File Name:

The filename must be requirements.csv. The filename is case sensitive.

b. Permissions:

Owner and Group (read/write), World (read)

c. Data Format:

The output format is standard CSV. All special characters in the document should be converted to a standard html equivalent code. Superscript and subscripted text should be identified using the HTML tags <super> and <sub> respectively.

d. Required fields

The document should have the following fields, those that are required are marked as such. The fields can be in any order and additional fields may be included.

- ID **(Required)**
  This is the unique identified for the requirement. It is formatted as described in the EIC Systems Engineering topology documents.
- WBS
  The primary WBS number that is associated with this requirement.
- Description **(Required)**
  This is the HTML formatted text of the requirement.
- Status
  This is a text string describing the status of the requirement.
- Comment
  This is an HTML formatted comment or comments about the requirement.
- rDate
  This is the date that this record was entered or last updated. The date format should be mm/dd/yyyy. If this field is blank, then the current date will be used.

  Note: If the record date is newer than an existing, identical record in the archive, then the older date will be used.
- TBD
  A text string containing TRUE or FALSE, indicating whether there are criteria within this requirement that are to-be-determined.

2.2 relationships.csv

This file contains the relationships between the current requirements. The file must meet the following criteria.

a. File Name:

The filename must be relationships.csv. The filename is case sensitive.

b. Permissions:

Owner and Group (read/write), World (read)

c. Data Format:

The output format is standard CSV. Since the fields only contain requirement IDs, there should be no special characters in any field.

d. Required fields

The document should have the following fields, all are required. The fields can be in any order and additional fields may be included.

- Parent **(Required)**
  This is the unique identifier for the parent requirement in the relationship. If a child has multiple parents, each relationship should be listed on a separate line.
- Child **(Required)**
  This is the unique identifier for the child requirements in the relationship. If a parent has multiple children, each relationship should be listed on a separate line.

2.3 subsystems.csv

This file contains the subsystem dictionary, which associates a name and WBS entry with each subsystem identifier. The file must meet the following criteria.

a. File Name:

The filename must be subsystems.csv. The filename is case sensitive.

b. Permissions:

Owner and Group (read/write), World (read)

c. Data Format:

The output format is standard CSV. If the *Name* field contains special characters, they must be converted to a standard html equivalent code. Likewise, any superscript and subscripted text should be identified using the HTML tags <super> and <sub> respectively. The *ID* or *WBS* fields should contain no special characters.

d. Required fields

The document should have the following fields, those that are required are marked as such. The fields can be in any order and additional fields may be included.

- ID **(Required)**
  This is the unique identified for the subsystem. It is formatted as described in the EIC Systems Engineering topology documents.
- Name **(Required)**
  This is the descriptive name of the subsystem. *Note: It is not necessary to provide a name that contains the entire context of the tree where this subsystem exists. A short name, that describes its specific purpose is sufficient. The context can be derived from its position in the tree.*
- WBS
  The primary WBS number that is associated with this subsystem.

2.4 archive.csv

This file will have been generated during the prior execution of the ReqModel application and contains all of the requirement records, as well as all of their variations that have been recorded over time.

a. File Name:

The filename must be archive.csv. The filename is case sensitive.

b. Permissions:

Owner and Group (read/write), World (read)

c. Data Format and Required Fields:

Identical to requirements.csv.

3. SOURCE PROCESSING

3.1 cron script processing

Source processing is initiated when the cron script detects that the file requirements.csv has been placed in the Data Sources directory. When the file is detected, the script will perform the following actions.

1. Check to see if the source files exist. If they do not, then exist immediately.
2. Check to see if /tmp/reqModelLock exists. If the file exists and is less than 1 day old, terminate the script because the data is already being processed.
3. If the file does not exist OR is more than 24 hours old, recreate /tmp/reqModelLock with the current time stamp and continue.
4. COPY the source files to the archive directory, leaving the originals in place. They should be renamed as follows:
   - requirements.csv becomes MM-DD-requirements.csv, where MM is the two digit month and DD is the two digit year.
   - relationships.csv becomes MM-DD-relationships.csv.
   - subsystems.csv becomes MM-DD-subsystems.csv
   - archive.csv becomes MM-DD-archive.csv
5. Run the ReqModel application using the configuration file located in the ReqModel application directory.
6. DELETE the source files requirements.csv and relationships.csv. Do not delete subsystems.csv or the newly revised archive.csv.
7. DELETE /tmp/reqModelLock.

3.2 ReqModel processing

The ReqModel application will create the webpages and directories as specified in the configuration file. The application is started with the command ReqModel -x /var/ReqModel/EIC.cfg.

Using the EIC.cfg script, the application will perform the following actions.

1. Create the master requirements page at https://eic.jlab.org/Requirements/index.html
2. Create the global requirements page at https://eic.jlab.org/Requirements/GLOBAL.html
3. Create the EIS requirements page at https://eic.jlab.org/Requirements/EIS.html
4. Create the ESR requirements page at https://eic.jlab.org/Requirements/ESR.html
5. Create the HSR requirements page at https://eic.jlab.org/Requirements/HSR.html
6. Create the IR requirements page at https://eic.jlab.org/Requirements/IR.html
7. Create the detector requirements page at https://eic.jlab.org/Requirements/DET.html
8. Create individual pages for each requirement in the tree below https://eic.jlab.org/Requirements/Requirements
9. Create the file archive.csv containing the current, active requirements and their prior versions at /sgroup/phys_div/Transfer/EIC-Requirements/archive.csv.

4. DOCUMENT DEPLOYMENT

In addition to the files updated and installed by the ReqModel application, the original requirements.csv file will also be copied to /sgroup/phys_div/www/eic/html/Requirements/source.csv. This file will be web accessible at the address https://eic.jlab.org/Requirements/source.csv .

Appendix 1: Crontab Entry

```
# Run the ReqModel application every hour on the hour
0 * * * * /var/ReqModel/bin/run.sh
```

Appendix 2: Cron Script /var/ReqModel/bin/run.sh

```
#! /bin/bash

srcPath="/sgroup/phys_div/Transfer/EIC-Requirements"

arcPath="$srcPath"/archive
reqFile="$srcPath"/requirements.csv
relFile="$srcPath"/relationships.csv
sysFile="$srcPath"/subsystems.csv
arcFile="$srcPath"/archive.csv
interval=$(date -d 'now - 1 days' +%s)
appFile=/var/ReqModel/bin/ReqModel
cfgFile=/var/ReqModel/bin/EIC.cfg
lockFile=/tmp/ReqModelLock

notifyCmd='sendmail -t "akers@jlab.org akers.walt@gmail.com"'

# -------------------------------------------------------------------------
# If the verbose flag is uncommented, then the script will emit
# a significant number of comments.
# -------------------------------------------------------------------------
#verbose=1

function printMsg () {
    if [ -v verbose ]; then
        echo ReqModel: $@
        fi
    }

function sendMsg () {
    echo -e $@ | sendmail -t "akers@jlab.org akers.walt@gmail.com watkins@jlab.org"
    }

# -------------------------------------------------------------------------
# Set the umask to ensure that the output files are editable
# by the owner and group
# -------------------------------------------------------------------------
umask 0002

# -------------------------------------------------------------------------
# Test to ensure that the srcPath exists, if it does not then an error
# should be reported (regardless of verbose flag), and the script should
# exit
# -------------------------------------------------------------------------
if [ ! -d $srcPath ]; then
    printMsg "ERROR: Source path $srcPath does not exist"
    sendMsg "Subject:Error: ReqModel Source Path \n\nSource path $srcPath does not exist"
    exit
    fi


# -------------------------------------------------------------------------
# Test for the existence of the source files.  If the required files
# do not exist them exit immediately.
# -------------------------------------------------------------------------
if [ ! -f "$reqFile" ]; then
    printMsg "$reqFile" does not exist, exiting...
    exit
fi
if [ ! -f "$relFile" ]; then
    printMsg "$relFile" does not exist, exiting...
    exit
fi
```

```
# ------------------------------------------------------------------------
# Test for the existence of the lock file. If it exists and is less than
# one day old, then exit without processing data.  Otherwise, delete the
# lock file if it exists and continue.
# ------------------------------------------------------------------------
if [ -f "$lockFile" ]; then
    fileTime=$(date -r "$lockFile" +%s)
    if (( fileTime <= interval )); then
        printMsg Removing expired lock file "$lockFile"
        sendMsg "Subject: Warning: ReqModel Lock File \n\nExpired lock file was removed, check
for errors on server."
        rm "$lockFile"
    else
        _hrs="$(( ("$fileTime - $interval") / 3600 ))"
        _min="$(( (("$fileTime - $interval") % 3600) / 60 ))"
        _sec="$(( ("$fileTime - $interval") % 60 ))"
        duration="$_hrs hours $_min minutes $_sec seconds"

        printMsg Lock file "$lockFile" will expire in "$duration";
        sendMsg "Subject: Warning: ReqModel Lock File \n\nLock file exists and will expire in
$duration."
        exit
    fi
fi

# ------------------------------------------------------------------------
# Create the lock file
# ------------------------------------------------------------------------
printMsg Creating lock file "$lockFile"
touch "$lockFile"

filePrefix=$( date +%m-%d- )
reqTgt="$arcPath"/"$filePrefix"requirements.csv
relTgt="$arcPath"/"$filePrefix"relationships.csv
sysTgt="$arcPath"/"$filePrefix"subsystems.csv
arcTgt="$arcPath"/"$filePrefix"archive.csv

# ------------------------------------------------------------------------
# Create the archive path if it does not already exist
# ------------------------------------------------------------------------
if [ ! -d $arcPath ]; then
    printMsg Creating archive "$arcPath"
    mkdir $arcPath;
    fi

# ------------------------------------------------------------------------
# Copy the relevant files to the archive before starting
# ------------------------------------------------------------------------
printMsg Creating backup files

cp -f $reqFile $reqTgt
cp -f $relFile $relTgt
if [ -f sysFile ]; then cp -f $sysFile $sysTgt; fi
if [ -f arcFile ]; then cp -f $arcFile $arcTgt; fi

# ------------------------------------------------------------------------
# Run the ReqModel application
# ------------------------------------------------------------------------
printMsg Executing ReqModel application
$appFile -x $cfgFile

# ------------------------------------------------------------------------
# Copy the requirements to a common location for web access
# ------------------------------------------------------------------------
cp -f $reqFile /sgroup/phys_div/www/eic/html/Requirements/source.csv

# ------------------------------------------------------------------------
# Remove the existing files once the process is complete
# ------------------------------------------------------------------------
printMsg Removing source files
rm -f $reqFile $relFile

# ------------------------------------------------------------------------
# Remove the lock file
# -----------------------------------------------------------------------
printMsg Removing lock file "$lockFile"
rm -f $lockFile
```

Appendix 3: Configuration File /var/ReqModel/bin/EIC.cfg

```
# =============================================================================
# [config] section:
#       This section contains configuration entries that are applicable to all
#       commands.  The following entries are defined.
#
#       error log= The path to the file that will be used for error reporting.
# =============================================================================
[config]
error log=/sgroup/phys_div/Transfer/EIC-Requirements/errors.log

# =============================================================================
# [active] section:
#       This section contains a line by line list of the paths to the CSV files
#       that represent active data.
# =============================================================================
[active]
/sgroup/phys_div/Transfer/EIC-Requirements/requirements.csv

# =============================================================================
# [archive] section:
#       This section contains a line by line list of the paths to the CSV files
#       that represent archive data.  These may be records that were deleted,
#       or it may contain older versions of current records.
# =============================================================================
[archive]
/sgroup/phys_div/Transfer/EIC-Requirements/archive.csv

# =============================================================================
# [links] section:
#       This section contains a line by line list of the paths to the CSV files
#       that contain the parent/child links between requirements.
# =============================================================================
[links]
/sgroup/phys_div/Transfer/EIC-Requirements/relationships.csv

# =============================================================================
# [systems] section:
#       This section contains a line by line list of the paths to the CSV files
#       that contain the subsystem identifiers.
# =============================================================================
[systems]
/sgroup/phys_div/Transfer/EIC-Requirements/subsystems.csv

# =============================================================================
# [actions] section:
#       This section provides a list of actions that will be performed with the
#   data that is specified in the preceding sections.  Each action
#   identifies an action sub-section that defines the commands and
#   parameters for the action.
# =============================================================================
[actions]
web
gbl-web
eis-web
esr-web
hsr-web
ir-web
det-web
dict
detail
csv

# =============================================================================
# [web] section:
#       This section contains configuration entries that are associated with
#       the genWeb command. These include:
#
#   command=  The command that will be executed for this action, in this
#             case 'genWeb'.
#
#       title=    The title that will be displayed at the top of the web page.
#
#       subtitle= The subtitle that will be displayed at the top of
#                 the web page
#
#       path=     The output path where the output files will be written. This
#                 may be a complete or relative path.
#
#       req path= The path for the individual requirement files WITHIN
#                 the output path.
#
#       index=    The name of the output html file (index.html)
#
#       baseURL=  The baseURL path for the web page, if any.
#
#   systems=  A comma separated list of headers that will contain the
#                 configuration variables for the different sub-systems that
#                 will be written to the web page.
# ==============================================================================
[web]
command=web
title=EIC Requirements
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
index=index.html
baseURL=
systems=GBL,EIS,ESR,HSR,IR,DET

[gbl-web]
command=web
title=Global EIC System Requirements
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
index=GLOBAL.html
baseURL=
systems=GBL

[eis-web]
command=web
title=Electron Injector System Requirements
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
index=EIS.html
baseURL=
systems=EIS

[esr-web]
command=web
title=Electron Storage Ring Requirements
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
index=ESR.html
baseURL=
systems=ESR

[hsr-web]
command=web
title=Hadron Storage Ring Requirements
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
index=HSR.html
baseURL=
systems=HSR
```

```
[ir-web]
command=web
title=Interaction Region Requirements
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
index=IR.html
baseURL=
systems=IR

[det-web]
command=web
title=Detector System Requirements
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
index=DET.html
baseURL=
systems=DET

# ==============================================================================
# [detail] section:
#       This section contains configuration entries that are associated with
#       the detail command. These include:
#
#       command=  The command that will be executed for this action, in this
#                 case 'detail'.
#
#       path=     The output path where the output files will be written. This
#                 may be a complete or relative path.
#
#       req path= The path for the individual requirement files WITHIN
#                 the output path.
#
#       baseURL=  The baseURL path for the web page, if any.
# ==============================================================================
[detail]
command=detail
path=/sgroup/phys_div/www/eic/html/Requirements
req path=Requirements
baseURL=

# ==============================================================================
# [dict] section:
#       This section contains configuration entries that are associated with
#       the dict command which creates a sub-system dictionary page.
#
#       command=  The command that will be executed for this action, in this
#                 case 'dict'.
#
#       title=    The title that will be displayed at the top of the web page.
#
#       subtitle= The subtitle that will be displayed at the top of
#                 the web page
#
#       path=     The output path where the output files will be written. This
#                 may be a complete or relative path.
#
#       index=    The name of the output html file (index.html)
#
#       baseURL=  The baseURL path for the web page, if any.
# ==============================================================================
[dict]
command=dict
title=Sub-System Dictionary
subtitle=Electron Ion Collider
path=/sgroup/phys_div/www/eic/html/Requirements
index=SubSystems.html
baseURL=

# ==============================================================================
# [csv] section:
#       This section contains configuration entries that are associated with
#       the csv command which a csv report of the current data.
```

```
#
#       command=  The command that will be executed for this action, in this
#                 case 'csv'.
#
#       active=   The output file which will receive the list of active
#                 requirements - if empty active requirements will not be
#                 reported.
#
#       archive=  The output file which will receive the list of archive
#                 requirements - if empty archive requirements will not be
#                 reported.
#
#       links=    The output file which will receive the list of parent/child
#                 relationships - if empty relationships will not be
#                 reported.
#
#       systems=  The output file which will receive the list of sub-systems -
#                 if empty sub-systems will not be reported.
#
# ==============================================================================
[csv]
command=csv
path=/sgroup/phys_div/Transfer/EIC-Requirements/
active=archive.csv
archive=
links=
systems=

# ==============================================================================
# [GBL] section:
#   This is a sample section that provides the configruation parameters
#   that would be needed to add the Detector sub-system to the output
#   web page as a top level entity. These include:
#
#       prefix=   The prefix of the sub-system; i.e. every sub-system that has
#                 this string at the beginning of its name, will be included as
#                 part of this section.
#
#       icon=     This is the icon that will be displayed at the head of the
#                 section and will be programmed to open and close the section.
#
#       title=    The title that will be displayed at the top of the section.
#
#       subtitle= The subtitle that will be displayed at the top of section.
#
# ==============================================================================
[GBL]
prefix=GBL
icon=icons/Requirements.png
title=Global EIC System Requirements
subtitle=Global requirements associated with the Electron Ion Collider.

# ==============================================================================
# [EIS] section:
# ==============================================================================
[EIS]
prefix=EIS
icon=icons/Requirements.png
title=Electron Injector System Requirements
subtitle=General, functional and performance requirements associated with the Electron Injector
Systems of the Electron Ion Collider.

# ==============================================================================
# [ESR] section:
# ==============================================================================
[ESR]
prefix=ESR
icon=icons/Requirements.png
title=Electron Storage Ring Requirements
subtitle=General, functional and performance requirements associated with the Electron Storage
Ring of the Electron Ion Collider.

# ==============================================================================
# [HSR] section:
```

```
# ==============================================================================
[HSR]
prefix=HSR
icon=icons/Requirements.png
title=Hadron Storage Ring Requirements
subtitle=General, functional and performance requirements associated with the Hadron Storage Ring
of the Electron Ion Collider.

# ==============================================================================
# [SHC] section:
# ==============================================================================
[SHC]
prefix=SHC
icon=icons/Requirements.png
title=Strong Hadron Cooling Requirements
subtitle=General, functional and performance requirements associated with the Strong Hadron
Cooling of the Electron Ion Collider.

# ==============================================================================
# [IR] section:
# ==============================================================================
[IR]
prefix=IR
icon=icons/Requirements.png
title=Interaction Region Requirements
subtitle=General, functional and performance requirements associated with the Interaction Region
of the Electron Ion Collider.

# ==============================================================================
# [DET] section:
# ==============================================================================
[DET]
prefix=DET
icon=icons/Requirements.png
title=Detector Systems Requirements
subtitle=General, functional and performance requirements associated with the Detector Systems of
the Electron Ion Collider.
```