

Complete Standards App 0.9.0.9001 - Reference Guide

On behalf of the Neuroinformatics and Biostatistics (NIBS) platform

Updated as of October 31, 2020

This reference guide helps you install and use the ONDRI Complete Standards App, with instructions on how to run the app, its limitations, and future updates.

Contents

1 Overview	2
2 Installation	2
3 Deployment	3
3.1 Download repository	3
3.2 Git Bash	4
3.3 Next steps after choosing 3.1 or 3.2	5
4 Running App	6
4.1 Preparation	6
4.2 Running Standards	9
4.3 Program Limitations	11
5 Future Development	11
6 Development, issues, and updates	12

1 Overview

The purpose of the ONDRI Complete Standards App is to run a standards check on the required components of a data package: the DATA file, DICT file, README file, METHODS file, and any additional files (e.g., MISSING file). The app will point to the precise location of the issue in files and throughout the package, allowing for curators to quickly identify where to make the fixes and adjustments prior to formal standards checks and onto outlier analysis by the Neuroinformatics and Biostatistics (NIBS) team.

2 Installation

1. Install R first and then RStudio. Please choose the correct installer carefully as it will depend on your computer's operating system.
2. Install Git (again please choose the correct installer carefully). During the installation process, you can leave all installation options to their default and original configuration. However, you can download Git in a different folder path if you wish.
3. Open RStudio and run the following commands in the RStudio console (the bottom left pane) to install the necessary packages if you have not already done so:

```
install.packages(xml2)
install.packages(shiny)
install.packages(tools)
install.packages(shinyalert)
install.packages(V8)
install.packages(stringr)
install.packages(plyr)
install.packages(shiny)
install.packages(DT)
install.packages(data.table)
install.packages(lubridate)
install.packages(shinyalert)
install.packages(shinyWidgets)
install.packages(shinyFiles)
install.packages(shinyjs)
install.packages(dplyr)
install.packages(varhandle)
install.packages(lubridate)
install.packages(readr)
install.packages("https://cran.r-project.org/src/contrib/Archive/rowr/rowr_1.1.3.tar.gz",
  repos = NULL, type = "source")
```

4. You will know that the packages listed above are loaded if the checkbox beside each in the Packages tab are checked as shown in Figure 1. Add the checkmark to each for any that are missing from the necessary packages.

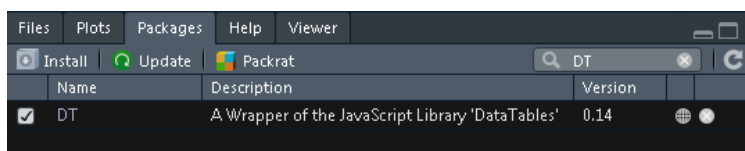


Figure 1: The “Packages” tab is located in the bottom right pane of RStudio.

3 Deployment

There are 2 ways to deploy the app: the first way is through downloading the repository directly off of GitHub and the second way is through Git Bash, We describe both options below.

3.1 Download repository

1. Go to the following link: https://github.com/ondri-nibs/standards_app
2. Click the green “Code” button as shown in Figure 2, and then click “Download ZIP”.

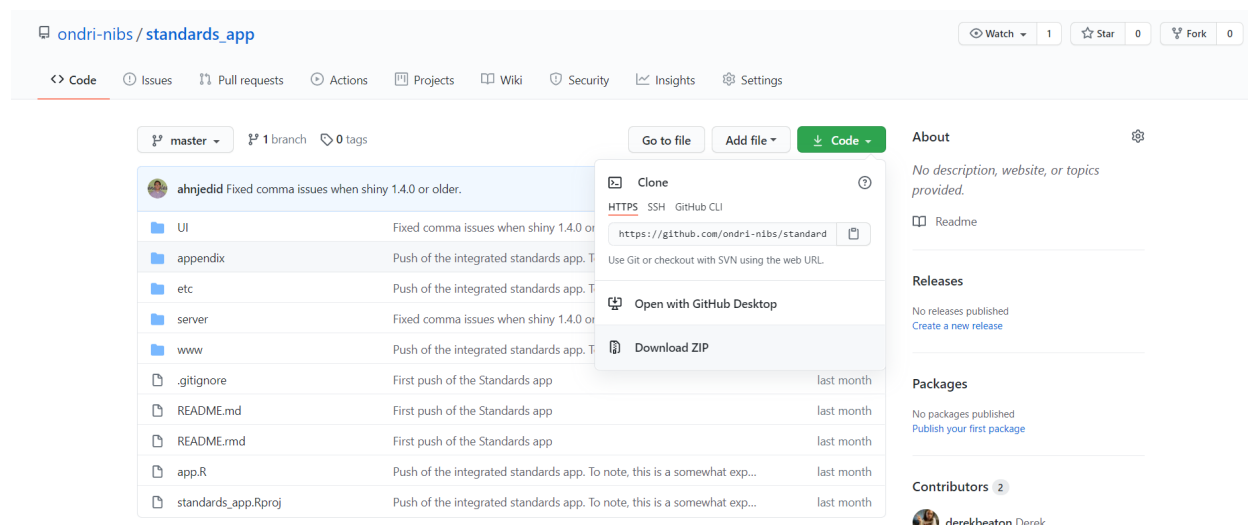


Figure 2: The repository is stored on the ondri-nibs GitHub organization page.

3. Browse to the directory that you saved the ZIP folder in, right click on the folder, and click “Extract All” as shown in Figure 3. Extract into a separate folder (not in the same directory as the ZIP folder). You may delete the ZIP folder afterwards.

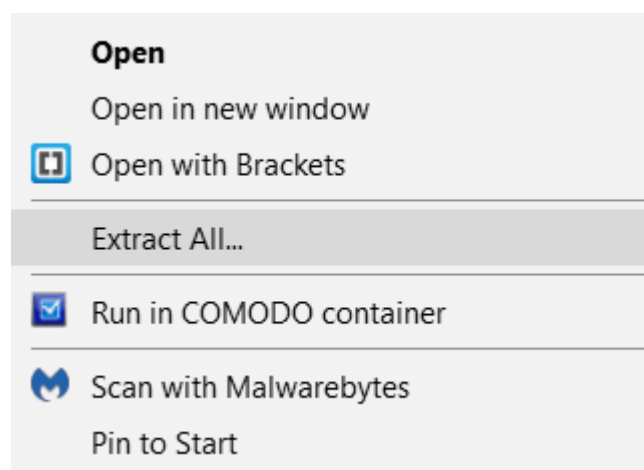


Figure 3: Please note that your drop down may look different depending on the operating system.

3.2 Git Bash

1. Right click on the folder that you wish to store the repository/program files in, and click on “Git Bash Here” as shown in Figure 4. The Git Bash application will pop up.

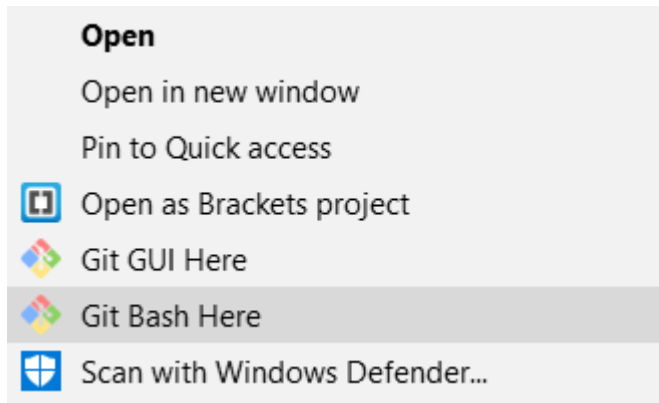


Figure 4: Please note that your drop down may look different depending on the operating system.

2. Clone the repository through Git by running the following command as shown in Figure 5:
`git clone https://github.com/ondri-nibs/standards_app.git`

A screenshot of a Git Bash terminal window. The title bar reads "MINGW64:/c/Jed's Work/CS Side Projects". The terminal shows the command `git clone https://github.com/ondri-nibs/standards_app.git` being executed. The output shows progress for cloning, enumerating, counting, and compressing objects. The prompt `$` is visible at the bottom.

```
Master@DESKTOP-OQVHAI4 MINGW64 /c/Jed's Work/CS Side Projects
$ git clone https://github.com/ondri-nibs/standards_app.git
Cloning into 'standards_app'...
remote: Enumerating objects: 172, done.
remote: Counting objects: 100% (172/172), done.
remote: Compressing objects: 100% (90/90), done.
remote: Total 172 (delta 79), reused 172 (delta 79), pack-reused 0
Receiving objects: 100% (172/172), 327.76 KiB | 1.16 MiB/s, done.
Resolving deltas: 100% (79/79), done.

Master@DESKTOP-OQVHAI4 MINGW64 /c/Jed's Work/CS Side Projects
$
```

Figure 5: The Git Bash terminal, which you can use only if you have installed Git.

3.3 Next steps after choosing 3.1 or 3.2

1. Open RStudio, go to File → Open Project, go to the repository folder, and double click standards_app (standards_app.Rproj) as shown in Figure 6.


Name	Date modified	Type	Size
appendix	2020-10-30 8:58 PM	File folder	
etc	2020-10-30 8:58 PM	File folder	
server	2020-10-30 8:58 PM	File folder	
UI	2020-10-30 8:58 PM	File folder	
www	2020-10-30 8:58 PM	File folder	
 standards_app	2020-10-30 8:58 PM	R Project	

Figure 6: standards_app.Rproj will be evident by the RStudio symbol on the left.

2. Open app.R in the RStudio file interface.
3. Click the dropdown of “Run App” in the top right of the file interface and select “Run External” as shown in Figure 7. This will run the app through your web browser rather than through a window.

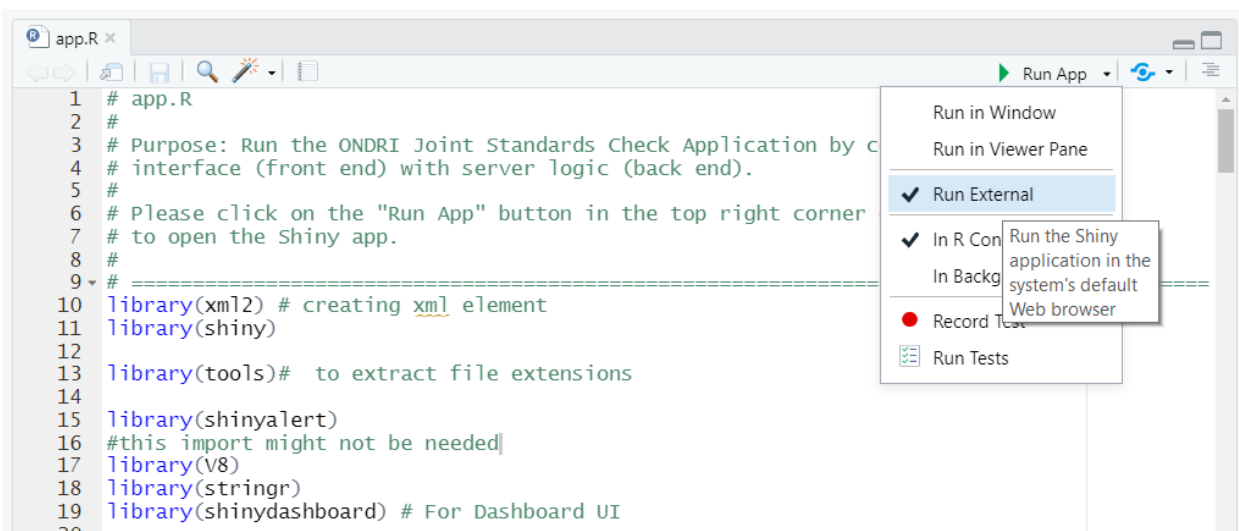


Figure 7: To open the dropdown, click the upside down black triangle to the right of 'Run App'.

4. Click “Run App” or type shiny::runApp() in the RStudio console.

4 Running App

Outlined in this section are details to prepare the app for running standards on a data package.

4.1 Preparation

1. Ensure that all files of the data package are in the same folder on your local computer. In addition, please download the participant IDs file into any folder. Ensure that you have the participant ID file relating to the appropriate study for your data.
2. In the Standards App, click on the button “Click to select a folder.”, and browse to the folder that you created. Then click the blue “Select” button in the bottom right corner as shown in Figure 8.

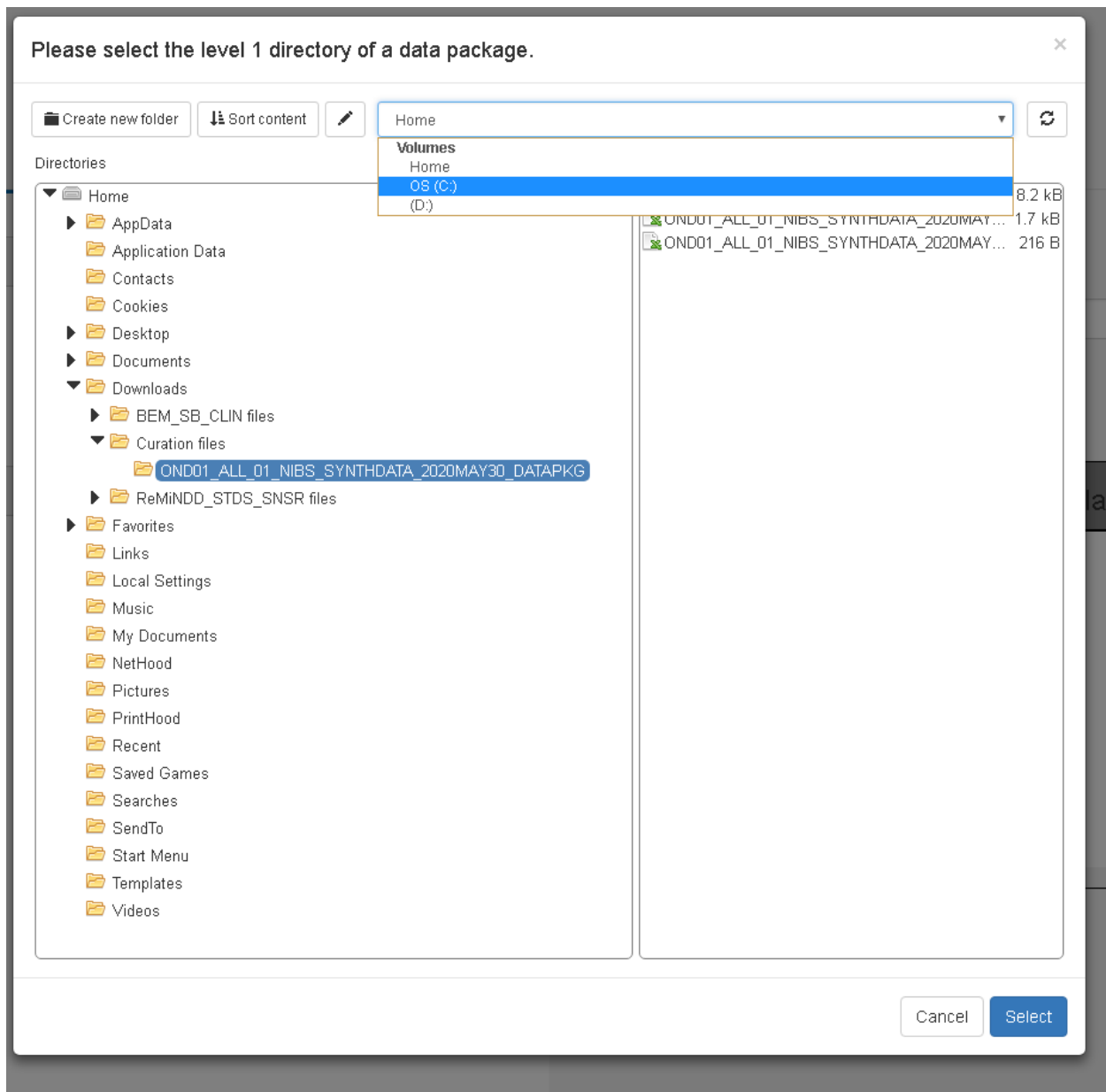
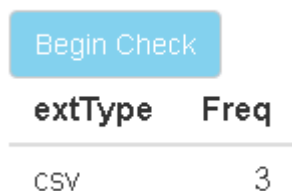


Figure 8: This window gives you access to both your home directory and your C: drive (on Windows).

3. The right-side panes will provide information about the loaded files. Once the package is selected, you can view what file extensions and their occurrence in the table which appears below the “Begin Check” button as shown in Figure 9.



The image shows a blue button labeled "Begin Check" above a table. The table has two columns: "extType" and "Freq". There is one row with the value "CSV" under "extType" and "3" under "Freq".

extType	Freq
CSV	3

Figure 9: The table will show you the number of occurrences for the different file extensions found in your data package.

IMPORTANT: The default directory chosen is your home directory. If you wish to access the C: drive (on Windows), please use the dropdown as shown in Figure 8 above.

4. Click on the button “Click to select an ID file.”, and browse to the folder that contains the ID file. Select the ID file and then click the blue “Select” button in the bottom right corner.
5. Once you have chosen the files that you want to run through the app, you will need to choose the appropriate study (under “Study Selection”) corresponding to the data that you will be checking (Figure 10). If the data you are running through the app requires different code checks from the quick select options, you can choose the “Custom” option under “Select Study” and add the codes you wish to check in the data package (Figure 11). If you would like to check your data package for non-specific items (such as blank cells, whitespace, etc), you can select the “Package only checks”.

Select Study

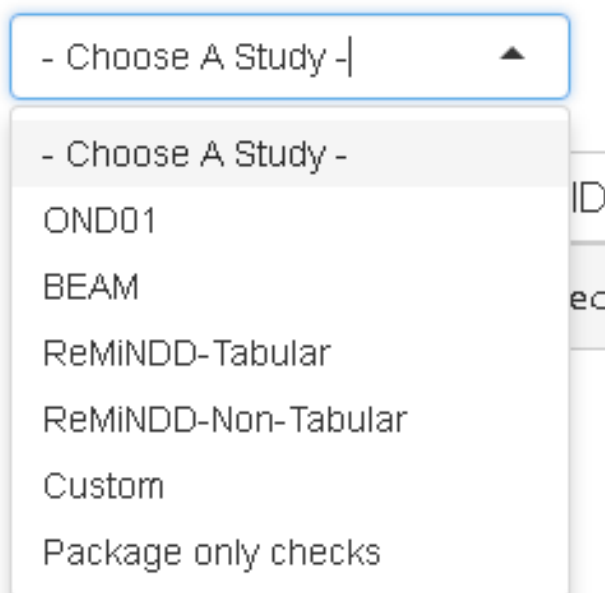


Figure 10: Study selection drop down menu.

Codes Form

Cohort Code

Visit Code

Platform Code

Site Code

Figure 11: In the top right side of the screen you can add custom study codes to be checked.

NOTE: You can view the codes that will be checked during the checking process in the table on the right side of the screen once a study has been chosen as shown in Figure 12.

Cohort Codes	Visit Codes	Platform Codes	Sites Codes
CNC	01	CLIN NIMG EYTK SDOCT GABL NPSY GNMC	BYC CAM SBH SMH TWH

Figure 12: The codes input table is found on the right side of the screen and it will let you preview the codes that will be checked during the process.

IMPORTANT: The app needs to have at least one item in all of the codes fields in the table to be able to run.

Congratulations, you are ready to start running standards on your data package!

4.2 Running Standards

1. Now that you are ready to check your data package, please click the “Begin Check” button.
2. When the button is clicked, a pop up will appear asking you to confirm that your data is tabular or non-tabular (Figure 13). If you have been directed to the wrong type (tabular/non-tabular), click “NO” and you will be redirected to the select page. If this is the case, you will need to check if you have accidentally selected the wrong package or if your tabular package contains sub directories causing the app to identify it as a non-tabular format. If it is the correct type, select “YES” to continue.

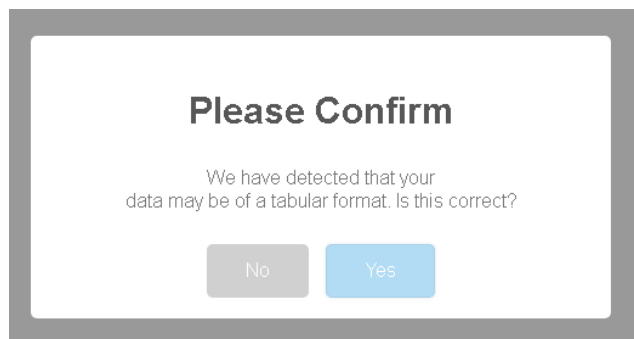


Figure 13: Example of a pop up when the program detects a tabular package.

3. Select the standards that you wish to check on the left hand side of the screen as shown in Figure 14 (by default, all standards are checked). Then click the appropriate buttons at the bottom of the screen to check the different files within the data package.

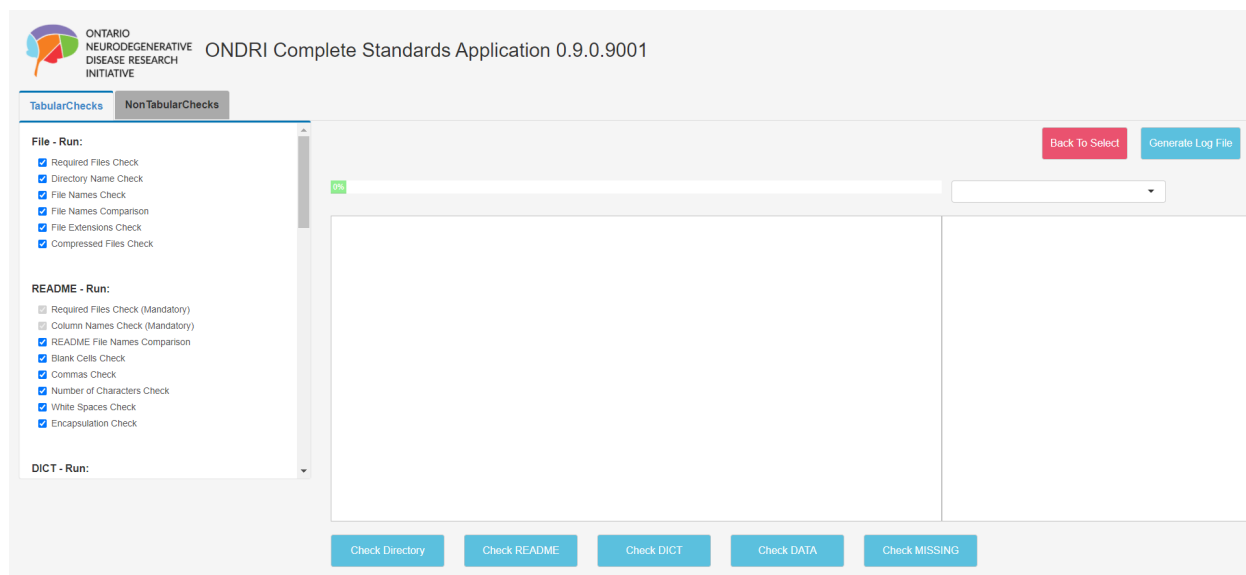


Figure 14: Interface for tabular packages.

4. For any checks that provide location of errors (such as blank cells), the precise locations are given through:

- Column titles and subject IDs in the DATA & MISSING file
- Column titles and column labels in the DICT file
- Column titles and file names in the README file

The dropdown next to the green progress bar displays the standard that has been violated and the column title where this violation occurs. The subject IDs, column labels, or file names that are listed in the right pane are associated with the violated values for this particular column.

5. For non-tabular data, you will click on the different tabs “Level1Checks”, “Level2Checks and”Level3Checks” as shown in Figure 15 and repeat the same process described in steps 3 and 4. This will check the sub folders of your data package.

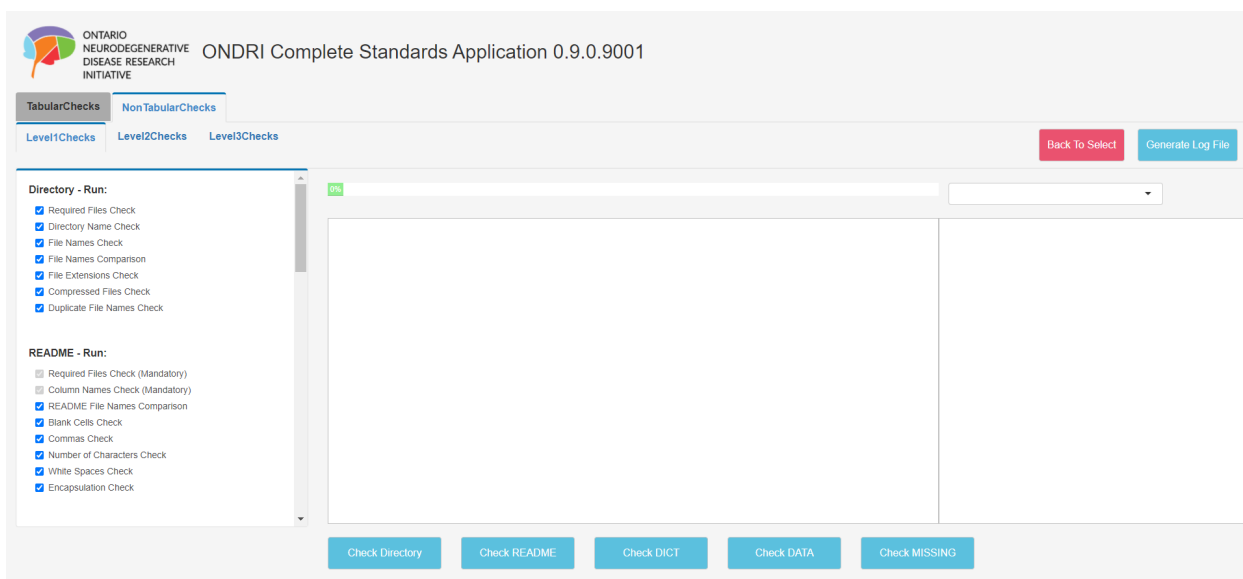


Figure 15: Interface for non-tabular packages.

6. Once you have performed all the checks you wanted to run, please click the “Generate Log File” button in the top right of the interface to generate a log file of all the checks that were run during the process. You will be asked to choose the location where this file is to be saved.

NOTE: At any point during the process, you can select the “Back To Select” button to return to the select screen. This will reset the app and all previous selections will be reset.

4.3 Program Limitations

Please note that a data package can pass standards despite errors when running the app. The following are common errors that get flagged and should be ignored in most cases:

1. **Blank Cells:** Flagging for blank cells always appears in the DICT file, as the VALUES column for SUBJECT and DATE column labels are usually blank. In addition, blank cells are allowed when a column is dependent on the values of another column. (Ex: If yes, explain why. If no, leave as blank.)
2. **Encapsulation:** Single quotes are allowed when used as an apostrophe.

The following are errors that pop up often, but should not be ignored. Rather, they are flagged to remind the curator and/or double check any values.

1. Missing codes M_TBC and M_OTHER are flagged as requiring special permission from the NIBS team. Please ensure that you receive this permission.
2. Data type MIXED is flagged as requiring special permission from the NIBS team. Please ensure that you receive this permission.
3. Date range: Date range is flagged to verify out of window dates. Once the dates have been verified, these messages can be ignored.
4. Number of characters: Any cells with over 200 characters are flagged and this is most common in the DICT file. While this is not a strict rule, please ensure that sentences are concise. After double checking, it is okay to ignore these messages.
5. **IMPORTANT: Commas are flagged often, and should not be ignored. All commas need to be removed and/or replaced with semicolons, as the file itself is comma separated.**

Furthermore, the app cannot check for spelling or grammatical mistakes, nor can it flag for dataset dependent errors such as adding up all values in columns and matching the sum with a TOTAL_SUM column.

IMPORTANT: Please note that a data package always has to be submitted to standards on LabKey. This is because the NIBS team does additional checks that the app cannot check for (such as variable naming).

5 Future Development

Potential features to look forward to:

- The ability to upload a codes spreadsheet which will autofill the codes input table.
- The ability to save codes inputted in the codes input table.

6 Development, issues, and updates

As the app is currently undergoing beta testing, patches will be pushed to GitHub as bugs and suggestions come in. Curators, RAs, and students will be notified when major updates become available. The version number will be updated accordingly in the app, as shown in Figure 16.

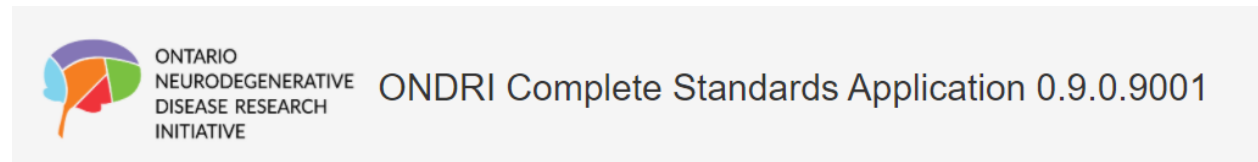


Figure 16: The version number will indicate whether you are using the latest version of the app.

If you chose to download the folder, simply download the folder again. If you installed Git, simply pull from the existing folder through Git Bash by running the following line: **git pull origin main**

For any bug reports or enhancement suggestions, please add an issue to the GitHub repository (https://github.com/ondri-nibs/standards_app/issues) so they can be tracked. Please assign Derek Beaton, select no milestone, and label it as either a bug (red) or an enhancement (cyan).

If you require any clarification or have technical difficulties with the app, please contact Derek at dbeaton@research.baycrest.org.