**Take Home Coding Assignment**

Author: Leonid Tochinski   
leonid@tochinski.com

Date: January 21, 2021

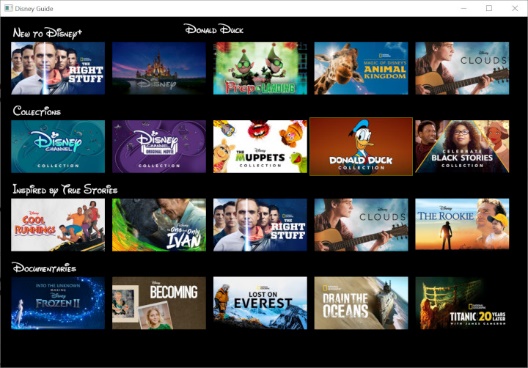
**Quick Start**

Extract *DisneyGuide.zip* to empty directory on Windows and run ***App\DisneyGuide.exe***

Internet connection is required.

**Description**

DisneyGuide Take Home assignment as a part of Disney Streaming Services – Connected Devices interview.  
It was developed by **“DSS - NCP Engineer - Take Home Exercise v5.pdf”** specification.

DisneyGuide application accesses master configuration json as well as dynamic configuration jsons  
extracts guide items information and renders in on screen.   
  
Collections are represented as rows, and collection items as columns.  
Keyboard keys (back/left/right/up/down) provide navigation on items,  
esc key used for exit.  
Selected item is enlarged and highlighted by frame, its title is displayed on the top of the screen.  
When there are more collections or collections items then presented on screen, additional items will be shown  
when try to navigate “outside” of screen.  
When image is not available, stock image is displayed.  
  
  
DisneyGuide is implemented on Windows using VisualStudo 19 using OpenGL and 3rd party open source libraries.  
makefile is provided for Linux builds.  
  
Application was tested on Windows 10, Ubuntu 20.04, Raspberry Pi 3B+ (RPI OS 3.5 Dec 2020).  
[Raspberry Pi 3B+ demo](https://www.dropbox.com/s/ay2z7g0b8qdel1r/PXL_20210125_023145757.mp4?dl=0)  
  
 **DisneyGuide Package Structure**

*src* – source code

*dependencies/bin* – runtime dependencies

*dependencies/include* – 3rd party include files

*dependencies/lib* – 3rd party include files

*dependencies/README.txt* – 3rd party libraries info

*App –* DisneyGuide app and its dependencies for Windows from dependencies/bin  
*DisneyGuide.sln  
DisneyGuide.vcxproj  
DisneyGuide.vcxproj.filters -*  Visual Source project files *DSS\_TakeHome\_Assignment\_Leonid\_Tochinski.docx –* this file

Also, source code is available from <https://github.com/leonid-tochinski/DSSGuide>

**Build Instructions**

Extract *DisneyGuide.zip* to empty directory.

On Windows:  
Open *DisneyGuide.sln* and build program with Visual Studio.

On Linux:  
On Ubuntu, install the following packages:  
***sudo apt-get install libcurl4-openssl-dev libjansson-dev libfreetype-dev libglfw3-dev***  
**cd <*your extracted dir*>/DisneyGuide/src  
make  
make install**

For both, Windows and Linux, run DisneyGuide app from **dependencies/bin** directory.

**DisneyGuide Architecture**Disney Class diagram



*disney\_guide* – core class. Provides business logic and aggregates other objects.  
*guide\_obj -*  wraps all the UI and OpenGL functionality. Base class for *disney\_guide  
decompress\_jpeg* – converts jpeg to RGB bitmap in memory.  
*text2bmp* - converts string to grayscale bitmap in memory.   
*curl\_http* - is curl wrapper for getting json and jpeg data from web server  
*json\_parser* - is used for parson json and accessing objects in hierarchy-like   
  
Architecture is event driven . Key presses are used as events.  
To save resources, only visible collection items are used.  
Additional items are downloaded when needed.  
  
Application is single-threaded for simplicity.

**3rd party dependencies**

Curl 7.74.0  
*https://curl.se/windows/*License: MIT-inspired (https://curl.se/docs/copyright.html)

Curl dependencies:  
libcrypto-1\_1.dll  
*https://www.dll-files.com/libcrypto-1\_1.dll.html*

libssl-1\_1.dll  
*https://www.dll-files.com/libssl-1\_1.dll.html*

Jansson 2.13.1  
*https://jansson.readthedocs.io/en/2.13/*License: MIT

GLFW 3.3.2  
*https://www.glfw.org/*  
License: BSD-like

FreeType 2.10.4  
*https://www.freetype.org/*License: BSD-like or GPLv2 by choice (https://www.freetype.org/license.html)

DevIL 1.8.0   
*http://openil.sourceforge.net/*License: LGPLv2

**3rd party License considerations**  
Most libraries are licensed by distribution-friendly BSD and MIT licenses.  
DevIL library is distributed as LGPL.  
In our case It is safe to use commercially as loaded module (.DLL, .so).  
  
  
**Artwork**

New Walt Disney Font  
https://www.fontspace.com/new-waltograph-font-f22088

Disney Image  
<https://www.insider.com/every-time-the-disney-logo-has-been-changed>  
  
**Future Improvements**1. To speed up UI at startup and save the memory, parsing of JSON should be done at the same time it is downloading.  
2. Configurable number of collections and items on the screen depending on windows size.  
3. Display items as they are downloaded  
4. Additional “predicted” item images should be downloaded in when user starts scrolling outside visible area