

Tsum Tsum Advanced Heart Sending

Basic Information

Welcome to my advanced heart sending script for Tsum Tsum. Currently the feature set is quite limited, but will expand over time as I create further iterations of the tool.

The features currently are as follows:

- Automatically claim hearts – The tool can claim all hearts, or can claim hearts individually in order to send hearts back to unknown players
- Automatically send hearts – The heart of the tool, of course. You can change in the settings whether or not you send to 0 Heart players!
- Connection Error Detection – Based roughly on the error detection from **nething4tc**, the tool will attempt to continue when a connection error is detected.
- Automatic timer – The tool has a timer that will start a new round of automatic sending periodically
- Settings – An integrated settings menu, or an editable .ini file, allowing the user to modify the delay between rounds, or whether hearts are claimed individually
- Activity Sensor – A background check detects mouse movement, and pauses the script to let the user interact with other windows.

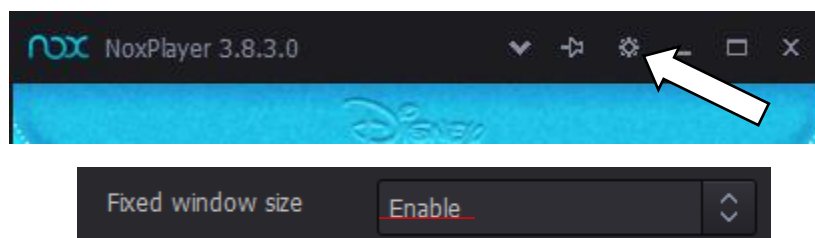
The connection error detection is currently unfinished. There are many errors that I need to account for, and many situations in which these errors can arise. Currently, I have only very basic detection, and the application has difficulty resuming afterwards. If anyone could assist in screen capturing and detailing these errors, I would very much appreciate the help.

If you would like to assist, please screen capture each step in order from when the error is received to when the error is gone, including a screen capture of the position in the application you end in when the error is gone. So if you're in the mail box when the error is shown, but you end up in the main page when you click okay, I would like a screenshot of the main page so I can figure out how to return to the mailbox when situations like this arise.

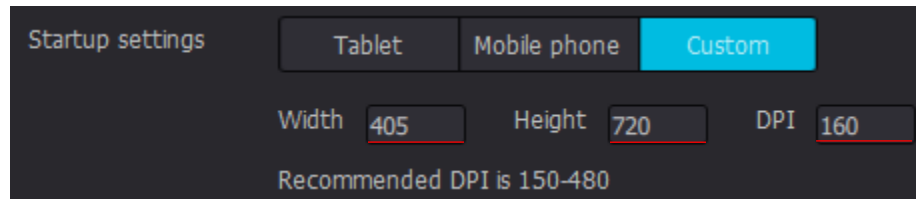
Basic Instructions

Step 1. Install Nox App Player – This part I will not detail heavily, as the standard installation of Nox is enough for the usage of this tool. Just download and install.

Step 2. Setup Nox App Player – For the tool to work, Nox must have a set size. The window cannot have its height or width adjusted beyond this size. First, start by turning on “Fixed Window Size” under the “Interface” tab of the System Settings for the Nox App Player



Next, change the “Startup Settings” under the “Advanced” tab of the System Settings for the Nox App Player. Set it to Custom, and change the Width, Height and DPI to match the illustration below.



Step 3. Save settings and restart the Nox App Player

Step 4. Install Tsum Tsum on the Nox App Player – Use the same method you would when installing from an android device. Use the Google Play Store to download and install the game.

Step 5. Run Tsum Tsum and log into LINE – Just as you would from your phone or tablet.

Step 6. Navigate to the Friend List screen on the game as shown



Step 7. Run the script via the included .ahk or the .exe

Step 8. Move the Nox App Player and the script window into positions that are convenient to you.

Step 9. Click “Start” on the TTAHS window.

User Interface

“Start” – This button starts the process of automatically sending and receiving hearts. The tool will begin by claiming all hearts, and then will navigate back to the main screen. After this it will scroll to the top of the list and begin sending hearts.

“Pause” – This button will pause the current process, and wait until you press the button again to resume activity.

“Stop” – This button attempts to stop the process. Sometimes, if the tool is in the middle of a check, there may be a delay before the process completely stops.

“n Rounds” – This is a display showing how many rounds the tool has completed. Each round is separated by a period of time which can be modified under File -> Settings...

“Log” – This is a log displaying certain actions of the tool. To some extent this is used for debugging purposes, though many actions are not logged currently in order to reduce output spam. Modifying the script will allow you to have certain actions logged, but I may not be able to assist with all changes made if an issue arises. When the thread is stopped, the log is dumped to the end of a log file in the root folder with the script!

“File” – Opens the file menu. You can select “Settings...” to change the round timer or set the tool to claim hearts individually. You can change the version of TsumTsum you are playing between INTL and JP. You can select “Exit” or press Esc when the “File” menu is open to close the tool.

F.A.Q.

Q: I’m using the JP version, and it’s getting stuck where it should be clicking “OK” to claim

A: *Not all of the JP images are included yet. For proper JP support you need to screen capture the JP images yourself and place the cropped pieces in the images/JP folder. In the root images folder you will find 8 images. Those images are the JP images currently missing. Sadly, I’m focusing most of my efforts into making the program function properly. The script, if the images are supplied, runs exactly alike on both versions of TsumTsum. The only time functionality would differ is a difference in the images supplied to the tool.*

Ask any questions or request support on any TTAHS thread on <http://www.reddit.com/r/TsumTsum> or contact: support@criticalgenesis.com

With any request for support, please include a screenshot of both the TTAHS tool and the Nox App Player, for review of the situation where the error occurred.

Version and Changelog

< Version b3.92 >

- Fixed a bug where the tool would fail to begin claiming images
- Implemented a fail-safe to multiple sections to try and recover the round when the tool fails to send a heart or claim a heart.

< Version b3.91 >

- Added another TTC option. There are two “categories” for time to click, Confirmation and Submit
- Added TTC to more situations in which a click might happen before an animation on the TsumTsum application finishes

< Version b3.9 >

- Complete rebuild of settings menu

- Created new kLib, containing a reusable library of GUI code
- Implemented first portion of TTC(Time to Click)
- Implemented setting to change the speed that the tool scrolls up at, and the delay on scroll checking
- Fixed a bug causing the script to launch a game session
- Hopefully squashed the living heck out of the bug causing the script to stop when preparing to confirm a heart claim.

< Version **b3.81** >

- Reimplemented window size checking to improve speed for image checks

< Version **b3.8** >

- Adjusted position of click point for individual claiming
- Made Pause button say "Resume" if thread is paused
- Fixed some inconsistencies that reduced performance when sending hearts

< Version **3.7** >

- Squashed bugs introduced by version 3.6

< Version **3.6** >

- Reimplemented error handling
- Added a cute little about page
- Rolled back a change on image checking that caused a version of Nox to be unable to use the tool
- Improved efficiency when claiming hearts
- Potentially squashed a bug where a heart wouldn't be fully claimed (Caused by no delay between two parts of the process)

< Version **3.5** >

- Fixed an issue with receiving heart requests.
- Individual Claim will send to heart requests, or deny heart requests, based on settings
- Restructured settings menu

Credits

I could not have done this without the amazing script by **nething4tc**. Seriously. Thank you for your script, as it taught me the basics of image and color matching using auto hotkey, and gave me a strong foundation with which to base my tool.