

Time's Up! v 1.2: a relative alarm

by James Steels
CIS 72351,661
INTERNET james.steels@rose.com

What is it?

... basically, a countdown timer--but with a few twists. Think of it like a kitchen timer, with punctuated intervals!

"Simmer for an hour and a half, stirring every fifteen minutes!"

"Laundry done in forty-five minutes..."

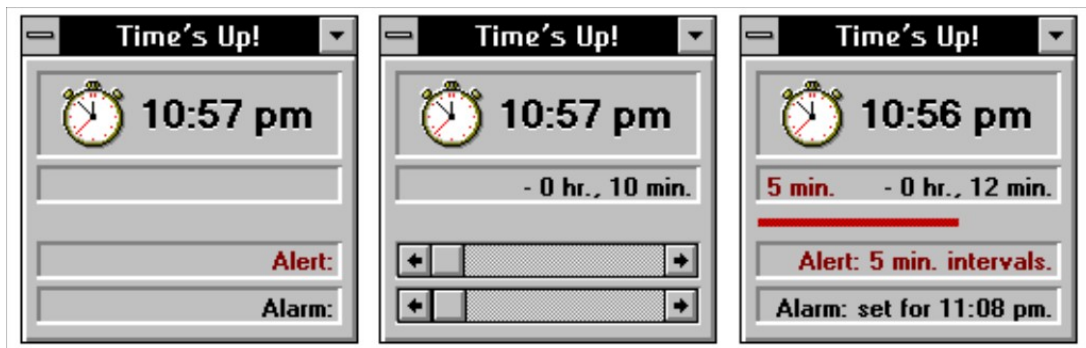
"Call back at fifteen minute intervals, until you get me..."

"Pizza delivery thirty minutes from now, or it's free." <g>

Time's Up! tracks it easily.



Fast Start: right click and hold on the stopwatch icon in the main display! The resulting map shows where everything is.



Time's Up! is entirely mouse oriented. With a few clicks, you can set up a fairly complex "pattern" of prompts, selecting an alarm time within a range of up to twelve hours, punctuating it with popup "alerts" at intervals of up to two hours.

Both the recurring "alerts" and the alarm can be used as standalone functions, or they can be used interactively.

How do you use it?

There are no buttons or menus in the main window, in order to reduce the screen footprint. The display is actually a cluster of mouse sensitive hotspots.

"Click" on any visible region, and you should get the response that's appropriate to that region...but "right click and hold" on the stopwatch icon in the top panel, and a map pops up that shows you where everything is. Simple, no? Play with it, a little, and you'll probably find it to be quite intuitive.

Both the alarm and the alert functions receive their input from popup scrollbars, which conveniently vanish after twenty seconds of inactivity, leaving a visible display:

- 1) recurring **alert information is dark red.**
- 2) **alarm information is blue/black.**

(Scrollbars can be forced to vanish immediately, with a right click on the stopwatch icon, or a right click on the "clear settings" panel. This last was added during development, to speed clearing the scrollbars from the display, but was handy enough to leave in the finished code. The right click will not disturb the entered values--a left mouse click will offer to clear the clicked value.)

Alarm information, in the second panel, counts down by minute to the target time.

Progress through an "Alert" sequence is displayed graphically, by an expanding time bar.

By default, entered information is immediately saved into the "timeup.ini" file in your Windows home directory. **If you exit the program, or need to exit Windows, any active information is restored the next time "Time's Up!" is started. Expired, missed alarms are simply deleted and ignored.** (Once again, use the kitchen timer metaphor: after the alarm period has expired, there is no response.)

Clicking again on a visible display clears that setting, and allows new input.

Four (not so obvious) things:

Sounds: after you first run "Time's Up!", you'll find two new entries in the **Control Panel's Sound** section, "Time's Up! Alarm" and "Time's Up! Alert". Select whatever .WAV files seem appropriate, if you wish to use something other than the default beep. You can shortcut using Control Panel: **a left mouse click on the stopwatch icon will bring up the Sound section immediately.** New entries may require restarting Time's Up! in order to take effect.

(NOTE: restored Alarm/Alert settings are announced by their respective .WAV files playing once each, on program load.)

Notifications: Time's Up! was intended to be unobtrusive, except when it has something to announce. Notifications are "always-on-top" for their duration, then the program subsides into the background. It should get your attention. (Somewhat to my surprise, this attribute works across PC-Tools for Windows' virtual desktops, pulling the program into whatever desktop display is currently active--a nice added benefit, I think. I was afraid that I'd have to arrange that.)



Exit: Time's Up!'s proper exit is actually an **icon, hidden in the current time display.** Wave your cursor over it, and it should pop to the surface, ready for use. If there is no program activity, it will exit the program cleanly, with no fuss.

However, if you are currently running either an Alert and/or Alarm, it will ask what you want to do with them, "save" or "kill".

Priorities: the program considers Alerts to be subsidiary to Alarms, a subset of the Alarm's time period. When an Alarm expires, the Alert expires with it. However, standalone Alerts are perpetual, until explicitly cancelled.

About:

Time's Up! is not quite shareware, not quite freeware--if you like it, feel perfectly free to use it. You could, however, slip a dollar or two, along with criticisms and suggestions, into an envelope and mail to:

**James Steels
561 Jarvis St., #21
Toronto, Ontario M4Y 2J1
CANADA**

- or contact me, at:

**CIS 72351,661
INTERNET james.steels@rose.com**

Trust me! Any currency that you have handy is acceptable! <g>

This was a Visual Basic learning exercise, my first actually, and I've learned a lot through its several revisions--probably violating quite few design guidelines in the process. <laughter> It also produced a utility/accessory, which I personally needed - I get too absorbed in the computer, when working, (cooking was really suffering) - and gave me ideas for quite a few more.

This was FUN!

Philosophy:

I have a criticism of much of the shareware that I see, which may be a little presumptuous of me, but here it is: the authors are often trying to do too much, to create applications that are too comprehensive and fill too many, too varied functions.

I don't need any more integrated clocks, thank you! I already have four. Telephone dialers? Five, at last count. Fixed moment alarms? About, eight, I believe. Program launchers? Far too many. ... and these are all too often grafted onto programs which don't require them, adding to overhead and interface complexity.

I intend to develop small, modular, focussed utilities and desktop enhancements.

Why not encourage me? <g>

Jim