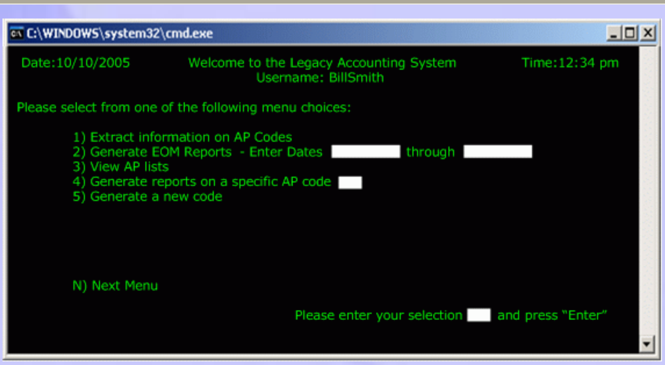
Cydonian Technologies, LLC

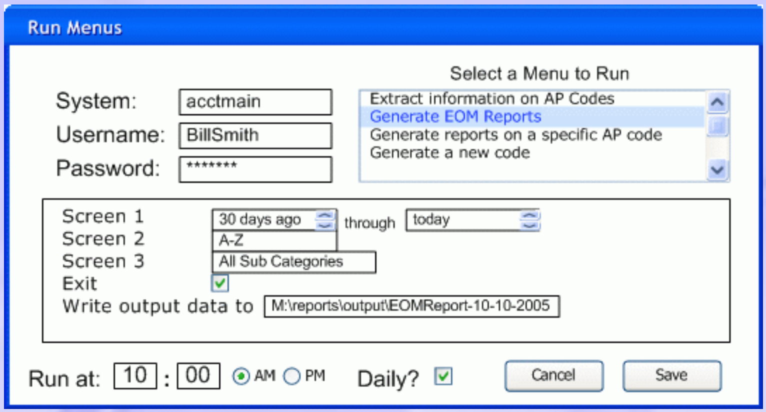
Case Study #2 – Migration of Telnet Applications to the Web

The Problem - The company has decided to automate existing applications. They purchased an "off-the-shelf" scheduling utility that gives them the ability to execute applications and commands at pre-determined times, and verify that the processes occurred. This is a tremendous improvement, with key applications no longer dependent on someone to run them at a specified time! Everything is going well, until an employee realizes that many critical functions in the accounting system cannot be scheduled, since they are run via a set of menus - not command prompts! Since this data is critical to many of the other scheduled applications, it becomes a stumbling block to automating the other processes. New automated processes are now failing because a staff member did not run the required accounting application at the required time. Because of a false dependence on automation, things are more chaotic now than they were under the old manual system. The automation project is seen as a failure!



```
C:\WINDOWS\system32\cmd.exe

Date:10/10/2005        Welcome to the Legacy Accounting System        Time:12:34 pm
                                 Username: BillSmith

Please select from one of the following menu choices:

        1) Extract information on AP Codes
        2) Generate EOM Reports  - Enter Dates          through
        3) View AP lists
        4) Generate reports on a specific AP code
        5) Generate a new code



        N) Next Menu

                              Please enter your selection    and press "Enter"
```

The Solution - Start with the end result in mind. In this case, the problem with the menu-based interface should not have been a discovery well into the automation process. Had it been identified as a concern, teams could have analyzed it proactively, rather than reactively.



Run Menus

System: acctmain
Username: BillSmith
Password: *******

Select a Menu to Run
- Extract information on AP Codes
- Generate EOM Reports
- Generate reports on a specific AP code
- Generate a new code

Screen 1: 30 days ago through today
Screen 2: A-Z
Screen 3: All Sub Categories
Exit: ✓
Write output data to: M:\reports\output\EOMReport-10-10-2005

Run at: 10 : 00 [✓ AM] PM Daily? ✓

While there may be a wealth of valid solutions, the best in this case was to create an application/process that simulates a user typing at a keyboard, and responds to the screen changes. This "superuser" application is fully interactive and navigates the screens, entering information and collecting data from the screens as it goes. As it traverses the menus, it documents what is sent and received for error-reporting purposes. The application can determine where it is in the menu system at any time, and if it ends up in unfamiliar territory (such as a change to the menu program), it can exit and alert staff to the problem. In the example screen at left, the user can enter all of the information in advance, and schedule a time to run. Values which might be dynamic during a run (such as today's date) can be collected from the system or from a database as defined. As shown in the example, the output can be written to a network drive, or sent to a web server or via email.

The result is a robust solution which would work cohesively with a scheduler and allow the company to retain its investment in the key system.