CSC318 Phase 1 : Project Proposal
Group 3

B.Hu - benj.hu@mail.utoronto.ca
D.Cho - dawing.cho@mail.utoronto.ca
R.Qiu - rex.qiu@mail.utoronto.ca
P.Bilodeau - philip.bilodeau@mail.utoronto.ca
M.Huang-Hobbs - maxwell.huang.hobbs@mail.utoronto.ca

Note: there is a significant potential overlap between solutions in our problem space and this semester's 301 group "Sensitive" (i.e. modifying the terminal to make it more accessible)

Problem Space

The problem space we intend to target is the initial learning experience of shell Command Line Interfaces (CLI). Though shell CLIs are powerful, they suffer from a steep initial learning curve. Users are expected to know a large repertoire of commands, arguments, options, and command composition techniques.

This problem stems mainly from the design of CLI interfaces. The primary use case where a Command Line Interface is more effective than a simple Graphical User Interface (GUI), is interfacing with complex systems that require many configuration options[1]. CLI interfaces also have the benefit of being faster than GUIs in many cases, as all interfacing is done through one focal point; thus, one's attention does not need to be drawn around a changing UI.

These traits make shell CLIs effective tools for systems administration. As a result, they are perceived and treated more as a tool, rather than an interactive program. Users are expected to read large manuals before trying to use a CLI, and something will inevitably go wrong. The debug/error messages that get displayed are targeted more to people with an existing understanding of the underlying systems, as opposed to new users just getting started with shells.

Despite being unfairly relegated to systems administration, many Command Line Tools are well suited to an average user's day-to-day operations. Many CLI-specific tools are faster, simpler ways of accomplishing common tasks such as file globbing and permissions management. Furthermore, many debug tools for common systems are usable only from a CLI (networks, etc).

Knowledge of a CLI is one of the main barriers between end users and developers / administrators. Moreover, most CLI tools operate on a small, shared design language that should not, in principle, be difficult to learn. Therefore, identifying and combating the main barriers to the initial learning and adoption of shells is a worthwhile task to pursue.

User Research Plan

We intend to sample subjects from the first and second year student population at UofT. After obtaining informed consent, the subjects will fill out a short survey to determine their level of familiarity with computers. Subjects will then be given a bash cheat sheet with descriptions of a few common commands, placed in front of a bash terminal, and asked to complete a list of tasks of varying difficulty.

Our primary focus should be determined by the largest overlaps between difficulties encountered across all skill levels, followed by those things specific to people familiar with computers, though not specifically terminals.

---

[1] The typical way of handling this in a GUI is with a series of dropdown menus, but as the number of configuration options rises, this becomes increasingly complex.