Heuristic Evaluation

By Joseph Rice

Team 6 - Disturbed WEPP Model

Undergraduate developers

- David Mohrhardt
- Joshua Hooker
- Jialin Lei
- Ashley Hauck
- Prakhar Keshari
- Mohammad Sajib Al Seraj

Graduate consultants

- Ankhita Pille
- Joseph Rice

Scientist

- Mary Ellen Miller

Project Summary

The Water Erosion Prediction Project (WEPP) is a hydrology and soil erosion model developed to predict soil detachment and movement. The aim of our application is to provide a simplified interface to these models and make them more accessible for educational purposes and/or users in the field.

UI Domain

Questionnaire / Survey with Results

Our user interface should be compared to informed surveying and questionnaire systems. Users will be expected to enter a significant amount of data and should be offered assistance in ensuring its correctness. Comparable software might include TurboTax.

UI Design Principles

Items and descriptions quoted from Nielson's Ten Usability Heuristics with additional notes addressing our application directly.

Visibility of system status (feedback)

- The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
- In our application, this might be displaying form progress at all times or indicating a model's synchronization status with the server.

Match between system and real world (metaphor)

- The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
- The user inputs for the Disturbed WEPP model are already broken down into a few categories, but it would be worth investigating if the inputs can be broken into smaller digestible portions that might fit an expected user workflow.

User control and freedom (navigation)

- Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
- In our application, this might mean saving partially complete forms and allowing users to exit then return at a later time.

Consistency and standards (consistency)

- Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

Error prevention (prevention)

- Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
    - Read full article on preventing user errors.
- In our application, presenting the appropriate input type, providing user input hints, performing server-side validation (perhaps in addition to client-side validation). Note: You can get clever with asynchronous techniques to get active validation from the server rather than waiting until form submission. Additionally, it might be possible to warn about valid but unusual/suspicious inputs. We should ask Mary about this.

Recognition rather than recall (memory)

- Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another.

Instructions for use of the system should be visible or easily retrievable whenever appropriate.
    - Read full article on recognition vs. recall in UX.

Flexibility and efficiency of use (efficiency)

- Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
- Mary's model can be used in to respond to lot of advanced queries; power users may want a higher-degree of control than the average user. We may be able to include these features to both appease these users and resolve complex errors.

Aesthetic and minimalist design (design)

- Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
- Our application deals with a lot of information, so presenting it in easily digestible portions is a key design concern. We need to present enough to allow the users to accurately and efficiently fill in the form, while keeping the cognitive load to a minimum.

Help users recognize, diagnose, and recover from errors (recovery)

- Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
- In our application, this could take the form of tips on how to correct invalid or erroneous input, perhaps leading the user to manual correction.

Help and documentation (help)

- Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Potential Usability Problems

Below is a list of usability problems and concerns discovered during the cognitive walkthrough.

1. The unlabeled inputs at the beginning of the form require users to either have prior knowledge of the model behind the system or make assertions and move on. This violates the memory principle.
2. The inclusion of the year at the beginning of the form may confuse users with simple use cases. While the model supports building a model over any period of time, the scientist indicated that the primary use case would be risk-analysis. This violates the metaphor principle. If we are to support additional use cases that might utilize those features, we should consider indicating their purpose to the user.
3. The “request inputs” and “query inputs” buttons purposes are not clear. This violates the memory principle. The interface should be intuitive enough for users to figure out how to use the application as they go.
    a. I suspect that this is an artefact of conflating the application’s purpose with the layout of the model. This is a critical design concern that violates the metaphor principle. The system architecture is largely irrelevant to the user, who might be using the app simply to decide if they should lay mulch out on the hill behind their house. Present what is useful and relevant to the user in terms that are consistent with their expectations.
4. There is no cancel button, this violates the navigation principle. Users may get to this point in the application and decide they wish to return to the homepage to access the “instructions” listed there.

Of the four screenshots analyzed during the cognitive walkthrough, two were of the existing Disturbed WEPP model interface and the other two provided little expectation of what the user interface might look like. More thorough prototyping will be necessary for Mary, Robert and myself to provide quality feedback.