

Heuristic Evaluation

App: CS Scheduling App

Scientist: Linda Ott

Team Members: (Developers) Alex McWilliam, Brian Conn, Collin Attard, Edward Fairchild, Kolby Swanson, Tyler Poirier, Will Sisson, **(UX Consultants)** Thanuja Maddali, Nikhil Nandala, Satyanarayana Velamala

App description:

The CS Scheduler for the Computer Science department is an intuitive application designed to simplify the process of creating class schedules. Users can easily select courses, specify preferences for times, locations, and instructors, and resolve conflicts with advanced conflict detection algorithms. The app offers flexibility through user-editable courses and conflict parameters, while also providing output generation options for easy sharing. Simple restrictions can be applied to the schedule, with warnings for potential conflicts, and users can access historical schedules for reference. Overall, the app ensures efficiency, accuracy, and user-friendliness in managing class schedules.

User Interface (UI) Domain:

In the CS Scheduler project, the UI Domain focuses on crafting a user-friendly interface for effortless scheduling. Designers will create visually appealing layouts and intuitive interaction patterns to guide users through the scheduling process seamlessly. By prioritizing clarity, accessibility, and responsiveness, the UI Domain aims to ensure that Michigan Tech College of Computing students can easily navigate the application and efficiently create their class schedules.

Heuristic Usability Principles

Visibility of System Status: Ensure users are always informed about the status of their scheduling actions, such as when classes are added, edited, or conflicts are detected.

Match between System and the Real World: Avoid using technical jargon or unfamiliar terminology when designing the interface and terminology so that they correspond with users' mental models of scheduling classes.

User Control and Freedom: Reduce the possibility of mistakes by giving users the flexibility to quickly move through the scheduling process, undo changes, and go back if necessary.

Consistency and Standards: Keep the application's layout, interaction patterns, and design elements consistent to facilitate user understanding and navigation.

Error Prevention: Put in place safeguards against common scheduling mistakes, like double-booking classes or neglecting prerequisites, by providing explicit warnings and validation checks.

Flexibility and Efficiency of Use: Allow users to customize their scheduling preferences, save preferences for future use, and expedite repetitive tasks through shortcuts or automation features.

Aesthetic and Minimalist Design: Aim for a simple, aesthetically pleasing interface that minimizes distractions and superfluous clutter while concentrating on important scheduling tasks.

Help and Documentation: Provide easily accessible help resources to aid users in navigating the scheduling process and troubleshooting issues, such as contextual help menus, report, or an extensive FAQ section.

Potential Usability Problems

In the CS Scheduler project, potential usability problems may include complex navigation structures leading to user confusion, inconsistent design patterns causing uncertainty, and hidden features impeding access to important functions. Additionally, unclear error handling and limited customization options could frustrate users, while accessibility barriers may exclude certain individuals from effectively using the application. Performance issues such as slow loading times can also hamper user efficiency. Addressing these issues through intuitive design, clear feedback mechanisms, and adherence to accessibility standards will be crucial to ensuring a seamless scheduling experience for Michigan Tech College of Computing students.

Critical Usability Concerns

Critical usability concerns for the Class Scheduler project include ensuring accessibility for all users, effective error handling to prevent critical scheduling mistakes, accurate detection and alerting of conflicting class schedules, optimal system performance during peak usage times, and providing adequate user support and resources. Failure to address these concerns could lead to issues, user frustration, and disruptions to students' scheduling processes. By prioritizing these critical usability aspects, the Class Scheduler application can deliver a seamless and user-friendly experience for students, enhancing their ability to create and manage their class schedules efficiently and effectively.