Proceedings of the Annual General Donald R. Keith Memorial Conference
West Point, New York, USA
May 4, 2023
A Regional Conference of the Society for Industrial and Systems Engineering

Complex Real-Time Human Resources Scheduling with Web User Interface Using Prioritizing Algorithms

Isabelle Hanchett[1], Emma Bachyrycz[1], Iris Bursey[1], Sara Hassan[1], Josef Thompson[1], Michael Purdy[2], and Raymond Romanczyk[2]

[1]Department of Systems Science and Industrial Engineering
Binghamton University
Binghamton, New York 13902

[2]Institute for Child Development
Department of Psychology
Binghamton University
Binghamton, New York 13902

Corresponding author's Email: ihanche1@binghamton.edu

Author Note: Isabelle Hanchett, Emma Bachyrycz, Iris Bursey, Sara Hassan, and Josef Thompson are senior Industrial and Systems Engineering students at Binghamton University, Thomas J. Watson School of Engineering. Dr. Raymond Romanczyk, Co-Director of the Institute for Child Development at the Department of Psychology, Binghamton University, and Michael Purdy, Coordinator of Technology for Institute for Child Development, are the team's capstone project advisors.

Abstract: The Institute of Child Development (ICD) at Binghamton University is uniquely complex in its staffing requirements. By offering extensive and personalized services to children with autism spectrum disorder, the ICD must ensure the correct number of properly trained staff are assigned throughout every school day to best serve each student. Since a broad range of human services professionals and paraprofessionals (e.g., psychology, special education, speech pathology, occupational therapy, etc.) all operate in one facility, each room's accommodations must also be taken into account. Due to these restrictions, the ICD is currently struggling to complete staff scheduling in a timely manner using Microsoft Excel. The team's main objective is to reduce the workload of this scheduling process by developing a dynamic web interface to integrate with existing related software and communication systems. By designing a more intuitive, assistive, and automated system, the team aims to significantly reduce the valuable senior staff time that could be dedicated to the many other roles they play in the success of students and their families.

Keywords: Staff Scheduling, Web Interface, CSS, HTML, JavaScript, PHP FileMaker

1. Introduction

1.1. ICD Background

The Institute for Child Development (ICD) is a NYS Education Department certified school located on Binghamton University's campus dedicated to providing services to children with autism spectrum disorder. The school was founded by Dr. Raymond Romanczyk in 1974. The ICD has a core staff of 82 with 7 additional contracted professionals and provides services to approximately 70 children at any given time within the age range of 2 to 11 years old. In addition to providing core education, it provides services to help these children develop their social and family relationships, emotional awareness, communication skills, and physical health. This includes social work, speech therapy, and an outdoor space for recreation. The quality of the services provided by the ICD are not available at traditional public schools and have a significant impact on both the students and their families. The students travel from as far as 80 miles each day to get to and from school. Each of these students have very different needs, so the ICD staff comes from a wide range of specialties.

Since the ICD offers such a broad array of personalized services for their students, this requires highly precise staff schedules each day. This has led to the ICD seeking assistance from our team in streamlining their very time consuming scheduling process by designing a web interface with ease of use and rapid distribution as the most important attributes for the user. Importantly, the system must be dynamic as schedules can change many times during the day in response to changing child and staff parameters. Thus, this real-time scheduling is very different from a simple 'reservation' style scheduling system.

2. Context Analysis

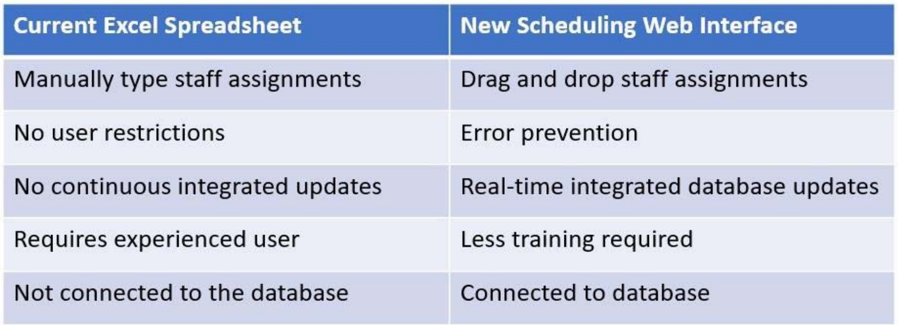
2.1. Current State

Managing the complexity of properly assigning staff resources to provide optimized child service schedules is a challenge at the ICD as there are many variables affecting staff and child availability that can change throughout the day, requiring immediate schedule changes. This is not only time consuming but also a vulnerability point for errors that could impact the quality of service the ICD is able to provide each day. The ICD currently uses an Excel planning sheet, which has evolved over time and is limited and difficult to use due to the number of variables and extensive rules that require a thorough understanding by the user. Administrative staff also must use a non-targeted, school-wide paging system to locate staff and children. This current system is inefficient as it takes hours to schedule every day, enables errors due to its complexity, and creates a need for repetitive and cumbersome communication. This system also greatly limits the speed of making adjustments and distributing finished schedules to the staff as each availability change requires the schedule to be manually re-worked. The updated system would allow for easier and quicker distribution of finalized schedules rather than relying on printing physical copies for all staff and repeating the entire process if any changes are necessary. Scheduling requires multiple senior level staff and takes time that can be better allocated to prioritizing the needs of the students.

2.2. Problem Statement

The ICD needs a more dynamic real-time scheduling system that can easily accommodate the unpredictable changes that each day may present. The goal of this project is to create a working web interface that allows users to assign ICD staff to specific classrooms in half hour increments in order to meet the specific student needs outlined in ICD's Rosetta Database. Instead of manually typing and adjusting schedule elements, this interface will allow easy user interactions such as dragging and dropping to avoid errors. The developed interface will also communicate with the ICD's primary administrative database that maintains required annual reporting information in order to extract the proper information as well as write user interactions back to the database to create an archived log for record keeping purposes. Following the completion of design analysis and opportunities for user feedback, the implementation of this system would reduce the workload for ICD staff very significantly and alleviate some of the time and pressure associated with producing highly personalized schedules each day.

Figure 1 illustrates the comparison between the ICD's current scheduling system, which is completed in an Excel spreadsheet, and the real-time scheduling web interface. The new scheduling web interface will focus on improving the efficiency of scheduling, as well as preventing user errors. Another major component of the new system is the connection to the ICD Rosetta Database. This allows for all the scheduling assignments to automatically be stored for record keeping, which is mandatory for compliance with federal and state regulations (*New York School Discipline Laws and Regulations by Category*, n.d.).



Current Excel Spreadsheet	New Scheduling Web Interface
Manually type staff assignments	Drag and drop staff assignments
No user restrictions	Error prevention
No continuous integrated updates	Real-time integrated database updates
Requires experienced user	Less training required
Not connected to the database	Connected to database

Figure 1. The ICD's current Excel spreadsheet system vs. the new scheduling web interface

The context diagram shown in Figure 2 summarizes the three main components that interact with the project. It shows how they interact with one another, the inputs required for the system, and the output it produces. For this system to function, it is required to have the data stores for color-coded staff responsibility level, classroom type, and staff availability. The first main component is the application, which is the visual interface and algorithms. The second is the objects, which refer to the defined data profiles for each specific staff member. Lastly, there is the ICD Rosetta Database. This is where the data is stored and updated. The application component has the written code, which creates the restrictions for the objects. The application also generates the scheduling interface based on the ICD Rosetta Database data that it imports. The object component interacts with the ICD Rosetta database because the database holds the data store. The main components interact in a way that allows for the output of the schedule assignments.

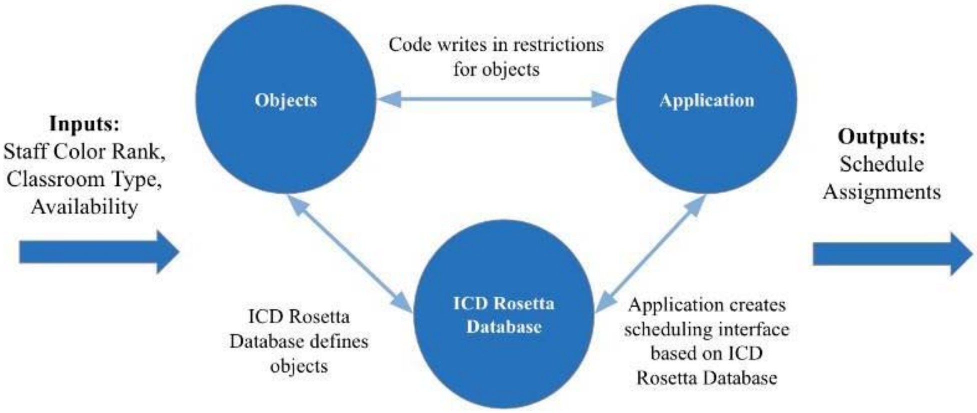

Code writes in restrictions for objects
Objects
Application
Inputs:
Staff Color Rank,
Classroom Type,
Availability
Outputs:
Schedule Assignments
ICD Rosetta Database
ICD Rosetta Database defines objects
Application creates scheduling interface based on ICD Rosetta Database

Figure 2. Context diagram of system components and interactions

3. Methodology and Design

As a continuation of a previous capstone project, the bulk of the work in the beginning stages was an analysis of the existing code. The originally developed code from the previous team did not contain any scheduling algorithms and served simply as a framework for its eventual implementation. There were four files provided in three different coding languages (CSS, JavaScript, and HTML). This required independently learning a range of software tools through online sources such as JavaScript documentation, as the team's previous experience was confined to MATLAB, Python, and R (*JavaScript Scheduler)*. The team also ordered three copies of "Web Development and Design for Beginners" to better establish the group's foundational knowledge (Webb, 2020).

In order to run and analyze the code, the team had to acquire several software resources. The first piece of software downloaded was a Virtual Private Network (VPN) to ensure the security of the information coming from the ICD Rosetta Database. For team members with a Mac, this meant installing Tunnelblick on their personal devices, and those with a PC installed OpenVPN Connect. Each team member signed a confidentiality attestation of student and staff information before being provided with personal configuration files to allow the proper access to the required data on machines outside of ICD. The team also installed MAMP and Visual Studio Code in order to successfully download and run the provided code. With this software, the team could collaborate in analyzing the code and understanding utilized functions and syntax. This is also how the team is able to make and test changes as new functions are added.

Two of the files are written in JavaScript. The first file is named "staffobject.js." This file works to extract the relevant data from the ICD Rosetta Database. The information within the database pertains to the specific qualifications of the staff members as well as the respective staff requirements for each of the classrooms. Each staff member has their own specific dataset that creates a profile of their individual information. The second JavaScript file is "main.js." This is the main engine of the program, controlling the functionality of the visual interactive interface as well as allowing for communication back to the database. Its key functionality is that it sends an output of changes made in the database back to the ICD Rosetta Database. This is important because each change made within this project's interface must be sent back to the database to keep an official record of the staff schedule. The ICD Rosetta Database is the platform that the state and federal government uses to ensure that the ICD is compliant with the staffing and scheduling laws.

The second file type is CSS and is named “style.css.” This file is responsible for creating the stylistic features such as the color, layout, and sizing of the elements viewed on the web page. This file is called within the JavaScript and HTML files to change the appearance of the interface as certain functions occur. The last file type is HTML, and the provided file name is “index.html.” This file’s purpose is to tie each file together to generate the visual interface in an interactive web browser.

After the initial provided code analysis was completed, the team was able to move on to creating and implementing the scheduling algorithms to transform this interface into a working tool that follows the correct restrictions and regulations. The algorithms communicate with the ICD Rosetta Database via PHP FileMaker. The scheduling requirements are then able to be obtained from the ICD Rosetta Database, and the user interacts with our system to assign staff to classrooms to satisfy all of the scheduling requirements.

4. Development and Testing

4.1. Database Communication

After making aesthetic changes in CSS to reflect the ICD’s colors and logo, the team’s first task was to complete the loop of communication between the ICD Rosetta Database and the web interface. The starting code used a call to the PHP Script in order to extract all of the staff names from the database and populate them in the interface. This is where the previous functionality stopped. The team then wrote JavaScript code to extract and reformat the date, classroom, staff identification number, and time period of each name that is “dragged and dropped” in the interface. This information is able to be collected based on the parent and child node of the element being manipulated.

From this, a string is sent back to the ICD Rosetta Database using a similar PHP call that allows the database to interpret and store the schedule assignments and any changes properly. This function allows the schedules to be quickly modifiable and keeps the system in adherence with record keeping regulations.

4.2. Improving User Interactions

For a better user experience considering the team’s knowledge of human factors, the team created a two-way time filtering system that can be triggered using either the drop-down or the time buttons as seen in Figure 3. This changes the CSS and functionality of each cell to restrict users to only schedule one time period and avoid double-booking staff.

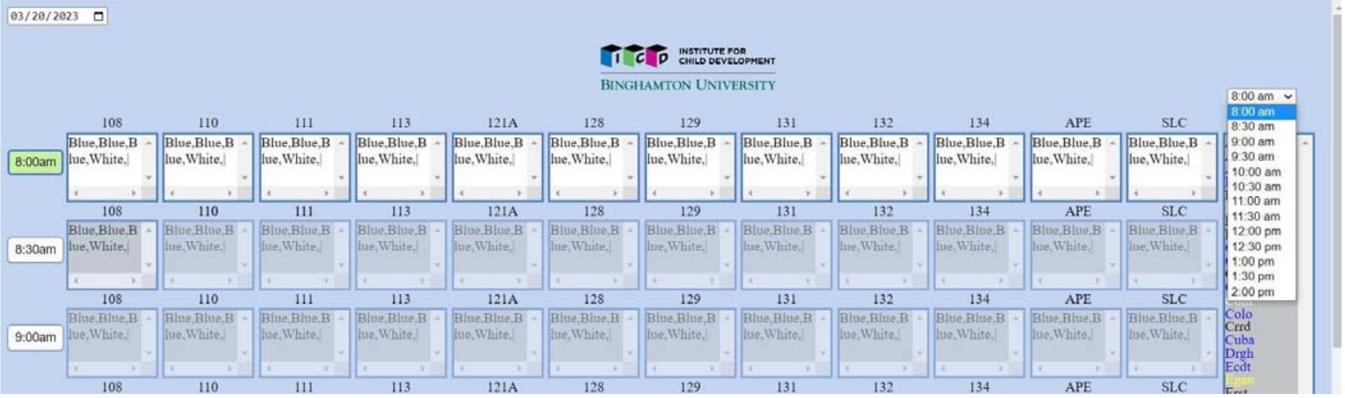
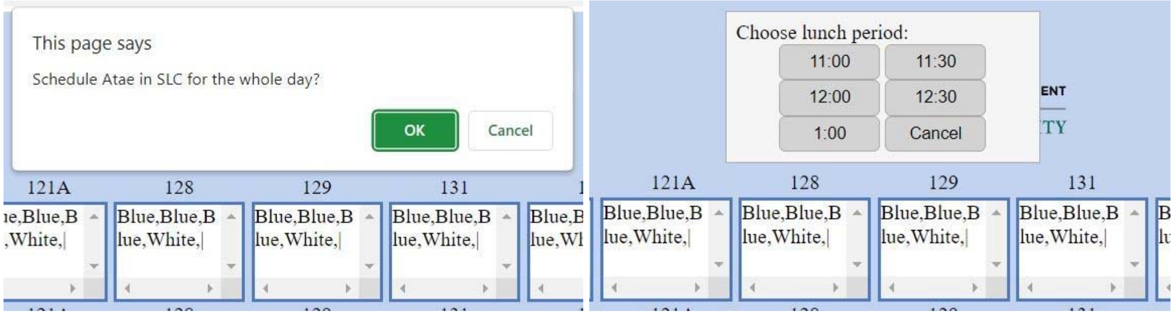


Figure 3. Time selection function to enable/disable cells and prevent double scheduling

The team also implemented an auto-scheduling option after feedback from an ICD staff member suggested this may be useful for the teachers that tend to stay in one classroom. The screenshots in Figure 4 show the prompt that appears if the user schedules any staff for the 8:00am time slot.



This page says

Schedule Atae in SLC for the whole day?

(a) User chooses to schedule the staff for the entire day

(b) User selects lunch period for that staff

Figure 4. Auto-Scheduling Function

4.3. Implementing Algorithms

After being provided a list of the rules currently followed by the ICD staff in building daily schedules, the team translated these rules into requirements, pseudo code, and then working JavaScript code. For debugging purposes, the team worked with our ICD sponsors in determining that a hard-coded array of classroom requirements would be used to check the functionality of the algorithms. Using this instead of pulling dynamic information from the database allows for better validation against a known outcome.

The scheduling requirements are primarily based on a staff member's "color ranking" (corresponding to their responsibility level - white, blue, yellow, red), the ratio of staff to students needed in each room, and the service needs of the students. In the near future, this information will be extracted from the database and tied to each classroom. Currently, the example array of required resources is `[Blue, Blue, Blue, White]` to represent three blue-level staff and one white-level (supervisory) staff member. By writing a loop that checks for similar elements between this array and that of the dragged staff name, the team was able to create scheduling restrictions on each matrix cell. If the selected staff member does not meet any of the specified needs for that cell, the program will produce an alert and kick the name back to the list. If it meets a requirement, the name will be placed, and the satisfied need will be removed from the requirement array in that cell. Once all cell requirements are met, the border will turn green to visually indicate completion. This can be seen in the uppermost APE cell shown in Figure 5.

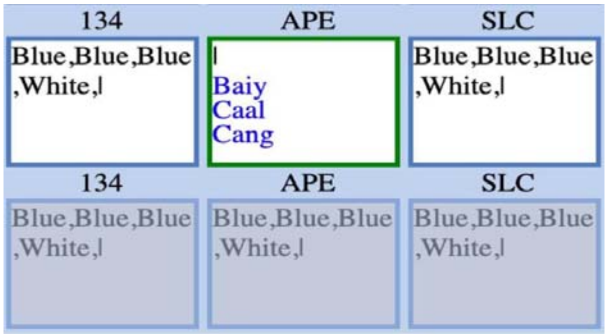


Figure 5. Green border for requirement fulfillment

5. User Feedback and Testing

To incorporate the voice of the customer throughout the team's design process, we periodically consulted several ICD staff. We started with our project sponsors, the Co-Director and Coordinator of Technology. The two provided us with a very thorough understanding of the ICD's needs and current system capabilities. By establishing a weekly cadence, their input aided the team in making decisions based on the information technology (IT) systems, federal and state regulations, and desired administrative improvements. Following this, another member of the ICD staff with experience in both IT and student instruction was able to join in on weekly meetings to incorporate an instructor's perspective into scheduling needs. Once the team developed a working model to demonstrate the core functionalities, the two senior staff responsible for scheduling were brought in for a user feedback session. Because these two staff members will be using the system the most, this session was an important opportunity for them to test the new interface, compare it to the previous interface, and suggest new elements for the team to incorporate. They provided integral design feedback, mainly on the aspects of their old system that they would like to see carried over (a separate indicator for staff breaks, different requirements for full-time and part-time staff, etc.). The team is addressing this feedback with the first priority of implementing a notes system and will reconvene for a follow-up session.

6. Impact and Future Steps

By establishing a more reliable and efficient method of scheduling, this project will support the mission of the ICD. The staff's focus can be shifted from scheduling issues to effectively serving children with autism spectrum disorder, through "compassionate partnerships and high-quality, person-centered, evidence-based practice" (*Institute for Child Development*, 2023). This system will create a more productive environment, as staff will be better equipped to mitigate issues stemming from unexpected changes. The updated system will also be able to provide individualized real-time staff schedules via hand-held devices already in place, as well as to classroom displays and kiosks. Since these are existing systems, this project allows functionality that is currently absent.

The implementation of this project also has a positive environmental impact because it will reduce the time spent on computers and allow for electronic transmission of schedules, which will dramatically reduce paper copies of schedules. Every day, the ICD prints up to 20 copies of the schedule on 3 pages and expects 3 revisions per day; these revisions also need to be reprinted to the 12 classrooms. This is 2,160 color copies per day and 453,600 copies per school year. By shifting to an electronic system, the ICD will significantly reduce their carbon footprint.

To continue this project, the team will dedicate the remainder of their time to facilitating user testing and feedback sessions with the ICD staff. This project is centered around the needs of the user, so it is of utmost importance that the opinions and concerns of all involved staff are the driving force behind the updated system. The team will also collect metrics such as median completion time to measure process improvement. This will help to justify the change to the new system and find any areas for further improvement.

7. References

Institute for Child Development. (n.d.). Retrieved from https://icd.binghamton.edu/mission-pg.html

Javascript Scheduler. *JavaScript Scheduler | DayPilot for JavaScript - HTML5 Calendar, Scheduler and Gantt Chart Web Components*. Retrieved from https://javascript.daypilot.org/scheduler/.

New York School Discipline Laws & Regulations by Category. (n.d.). *New York School Discipline Laws & Regulations by Category | National Center on Safe Supportive Learning Environments (NCSSLE)*, https://safesupportivelearning.ed.gov/discipline-compendium?state=new+york.

Webb, James. (2020). *Web Development and Design for Beginners*.