# 7Factor Staffing Tool #2

Major Qualifying Project 2023-2024

**Authored By:**

Jackson Lundberg

Matthew Adragna

Jordan Wecler

**Advised By:**

Professor Joshua Cuneo

**Sponsored By:**

7Factor Software, LLC

# Abstract

This paper presents the enhancements made to 7Factor's Software-as-a-Service (SaaS) platform, a Staffing Tool designed to streamline the company's time tracking and accounting functionalities. 7Factor, a software company specializing in custom DevOps and cloud-based solutions, previously relied on TimeIQ for time tracking. However, the limitations of TimeIQ, such as its complex user interface and lack of customization in billing rates, led to the development of an in-house solution tailored to 7Factor's specific needs. Our Major Qualifying Project (MQP) focused on further optimizing this Staffing Tool, primarily by integrating an automatic notification system. This system was designed to assist employees in organizing their time and keeping track of project deadlines without the need for manual tracking, thereby enhancing overall efficiency and workflow management. The paper details the design and implementation process of this notification system, the challenges encountered, and the milestones achieved, and provides recommendations for future work to continue improving the tool.

# Acknowledgments

The Staffing Tool team would like to acknowledge the following people for their guidance, support, and feedback throughout this project. Their contributions played an integral part in the direction and success of this project.

Professor Joshua Cuneo — *Major Qualifying Project Advisor*
Allen Brooks — *7Factor Director of Software Engineering*
Jeremy Duvall — *7Factor Chief Executive Officer*

Additionally, we would like to thank the other team of students who worked on the Staffing Tool for collaborating with us on this year's iteration of the project.

Nandita Kumar
Sean McNamara
Nicholas Borrello

Lastly, we would like to acknowledge past MQP groups who have worked on this project, for giving us a good foundation to build on.

| | | |
|---|---|---|
| Keval Ashara | Dyllan Cole | Panagiotis Argyrakis |
| Othniel Bondah | Noah Olson | Rafael Pimentel |
| Nicholas Li | Andrew Whitney | Nicholas Wood |
| | Evan Llewellyn | |

# Table of Contents

# List of Figures

# Executive Summary

## Notification System

One of our main goals for this project was to implement a notification system into the staffing tool to remind employees of upcoming deadlines and projects. Given the nature of 7Factor's workflow, a notification system would prove extremely useful to help employees organize their time and keep track of which projects they are working on, as well as what is being demanded from each project. Instead of having to manually keep track of deadlines for projects they are assigned to, the notification system allows employees to be reminded of deadlines as well as be able to find and view all recent reminders across all projects on the notification page.

## Skill-Match Search

Our reach goal for this project was to implement a Skill-Match Search feature into the application to help make assigning employees to projects easier for project managers. The Skill-Match Search feature introduces a page that allows for sorting employees by their skillset with the ability to quickly add them to a list for any upcoming project. This streamlines and hastens the process of creating a new project. This feature is also integrated with the notification system so that employees who are added or removed from a project will automatically have their project notifications updated accordingly.

## Clarity and Ease of Use

Another smaller goal of our project was for any additional UI pages or elements we create to be consistent with the overall theme of the application and to make sure that new features are clear and simple to use. To do this, we implemented components from other pages for familiarity while also creating new components in a similar theme to the rest of the application.

# Results

        As a result of this project, we were able to successfully implement a notification system that alerts employees of upcoming deadlines and projects. The notification system will only alert employees to notifications relevant to their specific work. Employees can easily navigate to the notification inbox within the Staffing Tool application, where they are able to view, search, and filter through all current notifications. Project notifications are automatically updated and created upon project creation and project changes. As notifications are ready to be sent out, they will appear on the notification inbox page.

        We were also able to implement a draft of a Skill-Match Search feature. Although the final iteration of this feature would need to include more in-depth skill processing, the current working version of this feature significantly accelerates and simplifies the process of adding employees to a project.

# Future Work

Our recommendations for future work from subsequent groups include the following:
- The integration of advanced analytics and accompanying features such as
    - Real-time updating of analytics
    - Customizable dashboard
- Allowing for user customization in
    - Dashboard views
    - Notification settings
- Security enhancement to prepare the workplace application use
    - Better encryption techniques
    - Increasing data storage security
    - Better access control system
- Data transfer objects
    - Making data transfer between the back end and front end more efficient
    - Making it so that only the data needed for the call is transferred

# Introduction

  7Factor, headquartered in Atlanta, Georgia, is a software company specializing in custom DevOps and cloud-based solutions on a contract basis. Their team, composed of remote members worldwide, focuses on creating, deploying, and advising on software for clients ranging from startups to large corporations. The objective of our Major Qualifying Project (MQP) was to enhance the existing Software-as-a-Service (SaaS) platform used by 7Factor. This platform is a Staffing Tool that combines time tracking and accounting functionalities, allowing the company to record and monitor employee time expenditure across various projects and assess associated revenues.

  To optimize this system, past groups, recognizing the limitations of 7Factor's past time tracking software (TimeIQ), implemented features that would help suit the needs of 7Factor more closely. TimeIQ, while functional, presents challenges with its complex user interface and lacks the ability to customize billing rates for individual engineer/project pairings. Past years' groups have made a simplified workflow for 7Factor staff by introducing a user-friendly interface and the flexibility to tailor billing rates based on specific engineer/project dynamics. The platform has a functional back end and database stack, as well as the basic front end developed in React.

  When trying to find further improvements to be made, we landed on the idea of an automatic notification system that would alleviate the need for 7Factor admins to send out emails or messages manually to their employees regarding project deadlines or housekeeping work. In this paper, we detail our design and implementation process, outline challenges faced, present the achieved milestones, and discuss recommendations we have for future project work.

# Background

## 7Factor

7Factor is a software consulting firm based in Atlanta, Georgia that specializes in designing, developing, deploying, and maintaining software solutions for its clients. The company-provided services include 'Project Execution' and planning, using 'Multimodal Partnerships' to construct tailored solutions and conducting audits to ensure compliance with industry standards such as SOC2, HIPAA, PCI, and other standards. (7Factor, n.d.)

At the core of 7Factor are their guiding principles, their '7 Factors': "Teach and Elevate", "Automate Everything", "Do No Harm", "Curiosity", "Equality and Diversity", "Good Things", and "Love" (7Factor, n.d.).

7Factor operates using small teams of 2-4 led by an engineering manager supported by a customer success lead for seamless client interaction. 7Factor has successfully collaborated with many well-known companies, including Delta Airlines, Aveanna Healthcare, and iVita Financial, and has provided quality solutions for numerous others (7Factor, 2023).

## TimeIQ

7Factor is looking for a time-tracking solution that can be used across the company to be easy and accessible to all employees. 7Factor uses time-tracking software to monitor employee log hours and productivity, as well as manage projects and billing to customers. This used to be provided by the paid time tracking software TimeIQ. However, as 7Factor grew as a company, they realized that TimeIQ cost too much money and required too much manual work to continue using their service.

The objective of the Staffing Tool is to be an in-house, cost-effective solution to fulfill the role of TimeIQ while implementing features that are specialized to the needs and workflow of the 7Factor employees.

# Past Accomplishments

In the development of the Staffing Tool for 7Factor, successive teams have contributed significantly, each building on the foundation laid by their predecessors. The earliest team established the cornerstone of the project by designing and implementing key back end and front end components of a web platform. This initial phase was pivotal in setting up a system to accurately track employee hours and manage staff utilization effectively, laying the groundwork for future enhancements.

Subsequent efforts by the following team focused on refining this platform, enhancing the user experience, and introducing analytical tools. This phase not only improved the software's usability but also incorporated significant back end developments to bolster new features and enhance security, thereby making it a more reliable and efficient tool for the company's operational needs.

Last year's team expanded the application's usability across various platforms, recognizing the growing importance of mobile and tablet compatibility. They developed a new user interface that aligned with the desktop version's theme, extending the software's functionality and accessibility to modern mobile and tablet displays. This adaptation significantly enhanced the software's versatility and user-friendliness.

Each of these phases, marked by distinct yet complementary focuses, has been instrumental in evolving the software to its current state, setting the stage for ongoing and future enhancements.

# Methodology

## Agile Development Methodology

Agile methodology in software development is an incremental approach that emphasizes collaboration, flexibility, and customer satisfaction (Cohen, Lindvall & Costa, 2004). We meet as a team for scrum meetings three times a week and organize our efforts into three-week sprints. At each meeting we discussed what progress we have made, and any issues we have run into, as well as reprioritize remaining tasks that need to be done.

## Development Tools

The development tools selected for this project were instrumental in its execution. They supported various aspects of the project, from planning and design to coding. These tools were chosen for their ability to enhance collaborative efforts, keep the project on track, and aid in the visualization of ideas, ensuring a cohesive and efficient development process.

### Gantt Chart

The project employed a Gantt chart to effectively manage and distribute the workload over two seven-week terms. A Gantt chart, a project management tool that displays activities against time, is instrumental in visualizing project schedules and assessing the duration of various tasks (Kerzner, 2017). In this project, the chart outlined four sprints, two per term for the main objective and one per term for a reach goal. Each sprint spanned 3-4 weeks, concluding with a demo to 7Factor. This structured approach facilitated continuous progress monitoring and ensured adherence to the project timeline. Regular bi-weekly meetings further supported this methodology, enabling efficient task completion and collaborative progress evaluation (Lock, 2014).

## Jira

Jira is an issue-tracking software that helps to create, track, and organize tasks. Jira is very helpful in our project, as it allows us to easily create a list of task items that need to be done, assign each task to a member of our team, and track the status of the task as we progress through the sprint. Another great feature of Jira is the customization it allows; for instance, we were able to set custom status markers for our tasks (To Do, In Progress, Ready for Team Review, Done) which helps us organize our progress and emphasize collaborative team reviews for each task before we can mark it as complete.

## GitHub

GitHub is a web-based platform that uses the Git version control system to organize and track changes in source code. GitHub is particularly useful for our project, as it also allows for remote collaboration and branching to protect tested, working code from new changes that might not be ready to use. Our project is broken down into three submodule repositories: database, API, and front end. These submodules are then added together into a separate repository that holds all three to keep an updated copy of the entire application.

## IDEs

For this project, IntelliJ and VSCode were chosen as the main Integrated Development Environments (IDEs). IDEs are software tools that help programmers develop software more efficiently (Farrell, 2015). IntelliJ is favored for its advanced code analysis and refactoring features, which are particularly useful for complex software projects (Flanagan, 2018). VSCode is appreciated for its lightweight design and flexibility, offering a wide range of customizable options (Fitzgerald, 2017). These IDEs were selected because they provide user-friendly interfaces and organize files effectively, and they can be tailored to different programming needs, making them ideal for developing the Staffing Tool.

## Figma

In the first phase of the project, Figma was utilized for initial designs. Figma is a cloud-based tool known for facilitating collaborative design work and enabling the creation of

detailed design prototypes (Thoring et al., 2021). It was employed to develop initial mockups for key parts of the Staffing Tool, such as the Notification page (as seen in Figure 1), Skill-Match page, New Project dialog box (as seen in Figure 2), and both taskbars (as seen in Figure 3). The use of Figma allowed our team to effectively visualize and refine the user interface before actual development. This approach of using Figma for prototyping and iterative design was instrumental in enhancing the design process's efficiency and outcome (Waloszek, 2018).
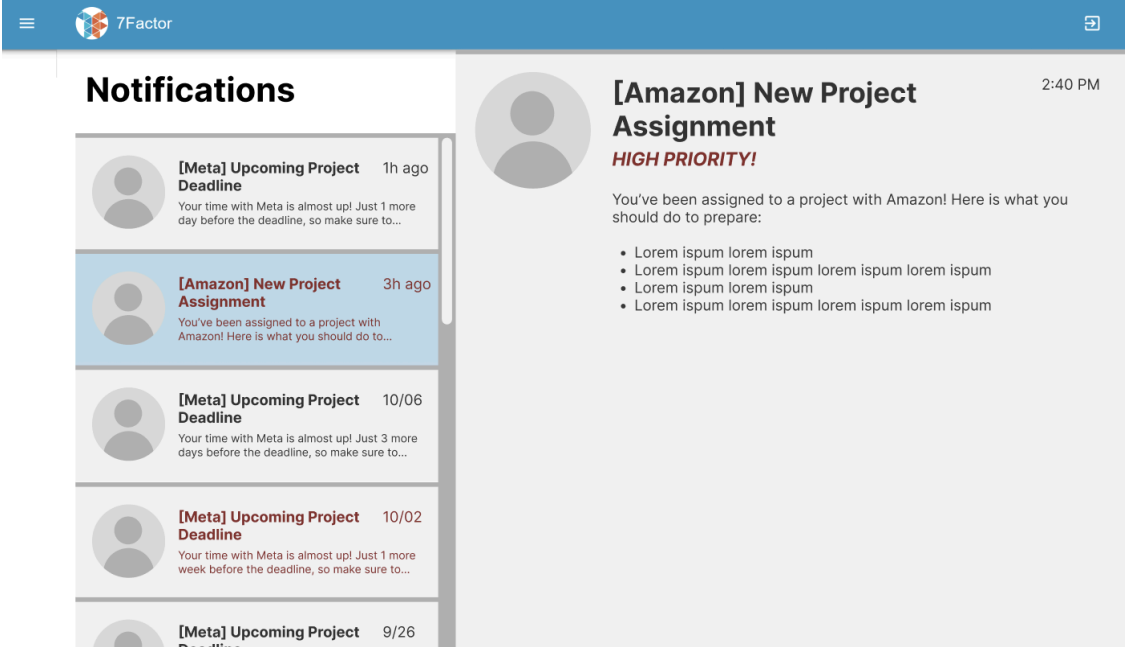


**Figure 1:** Initial UI mockup of the Notification page, created in Figma.

**Figure 2:** Initial UI mockup of the changes made to the New Project dialog box, created in Figma.
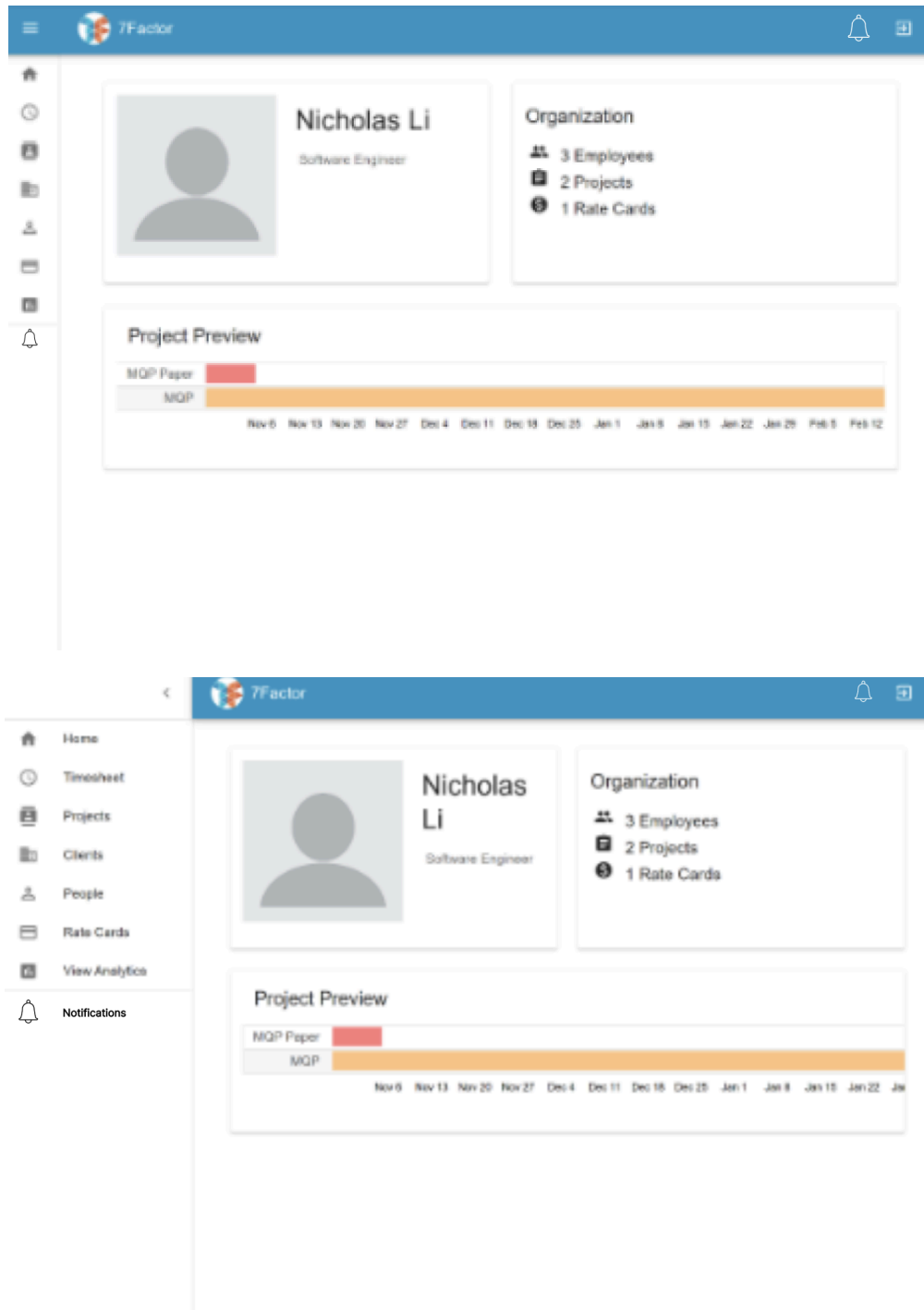
**Figure 3:** Initial UI mockup of the Taskbar and Menu bar in the opened and closed state, created in Figma.

# Front End Libraries

The project's front end development was driven by React and the MUI library. React's capabilities in building dynamic and responsive user interfaces were crucial in enhancing the user experience. The MUI library, known for its comprehensive set of design elements and tools, was utilized to structure and style the front end.

## React

React — a widely utilized JavaScript library — plays a central role in constructing user interfaces within the Staffing Tool application. Developed by Facebook, React is distinguished for its component-based architecture, which enables the development of complex user interfaces through the assembly of smaller, reusable components (Gaikwad, 2018). This modularity not only simplifies the UI construction process but also enhances maintainability and collaboration, as developers can work on individual components without disrupting the overall structure (McFarland, 2021). Moreover, React's virtual DOM feature ensures efficient updating and rendering of components, making it highly efficient for dynamic web applications (Banker, 2020). The choice of React for this project was driven by these capabilities, ensuring a user interface that is both robust and scalable.

## MUI Library

The Material-UI (MUI) library, a prominent React component library, was essential in shaping the front end of the project. Known for its comprehensive range of customizable HTML components, MUI enabled a tailored approach to interface design, aligning seamlessly with the project's specific needs (Pavithra, 2020). These components formed the backbone of the user interface, significantly enhancing its usability and interactive capabilities. MUI's rich selection of icons also played a crucial role, contributing to a consistent and aesthetically pleasing design and thus enriching the overall user experience. The utilization of MUI was critical in striking an optimal balance between functional effectiveness and visual refinement in interface development (Steinmetz & Wattenberg, 2019).

# Back End Libraries

For the back end development, Spring Boot was employed. Spring Boot streamlined the process by automating many routine tasks associated with setting up and managing the database. The database itself was constructed using PostgreSQL, a popular SQL database format. The choice of PostgreSQL was influenced by the uniform structure of the data stored for the Staffing Tool, making it a more suitable option than NoSQL alternatives.

## Spring Boot

Spring Boot, a Java-based framework, serves as the backbone for the back end architecture of the Staffing Tool application. Spring Boot is widely recognized for its three core features — simplified dependency management, streamlined deployment, and autoconfiguration (Heckler, 2021). Leveraging the benefits of Spring Boot's capabilities enhances the efficiency of development, enabling seamless integration of various modules within the application. Spring Boot's use in this application allowed for the creation of a clean, high-performing, and organized back end codebase.

## PostgreSQL

PostgreSQL is a robust open-source object-relational database management system that the Staffing Tool uses to handle persistent data storage. PostgreSQL is known for its performance, support for complex data types, extendibility, and adherence to SQL standards (Obe & Hsu, 2017). Leveraging PostgreSQL's capabilities ensures efficient data management, reliable transaction handling, and streamlined data retrieval within the application. Its seamless integration with the rest of the back end makes it fundamental for the back end functionality of the Staffing Tool.

## Docker

Docker, a leading containerization platform, plays a pivotal role in the deployment of the Staffing Tool Application. As a containerization platform, Docker encapsulates applications and their dependencies into containers, ensuring uniformity across various environments and streamlining deployment processes. Docker's ability to create standalone containerized

applications builds a layer of isolation in software that leads to software being more reliable, scalable, and easier to develop for teams (Kane & Matthias, 2018). In the context of the Staffing Tool Application, Docker's role is to containerize both the back end and the front end in Linux containers which allows for the program to run on almost any system with minimal setup.

# Results

## Feature Additions

At the onset of the project, we determined that our objective would be to expand the Staffing Tool's usage capabilities. To achieve this goal, we generated ideas for potential features, and after narrowing down the list, we set a primary goal and a reach goal: implementing a notification system and a skill-match search page respectively.

Our primary goal was to implement an automatic notification system that could remind employees about project-related tasks or administrative duties. This would ensure that employees are made regularly aware of their responsibilities, and would alleviate the need for manual reminders between project teams, project managers, and clients.

Our reach goal was to create a new page that would allow administrators to search for employees based on their predefined skills, and then assign them to a project team based on their skill set, allowing for quicker team assembly. This was a secondary focus for the project since the notification system was the main priority.

## Notification System

The following section outlines the additional functionality brought to the Staffing Tool with the Notification System. First, the process by which an administrator would automatically create and manage notifications for employees will be shown. Then, the process by which an employee would view and manage these notifications will be shown.

### Administrator Usage

When an administrator creates a project through the project creation dialog, there are now toggles that display the different types of notification events that can be enabled or disabled for that project. These include reminders about project start dates, demos, deadlines, and reassignments (as seen in Figure 4).
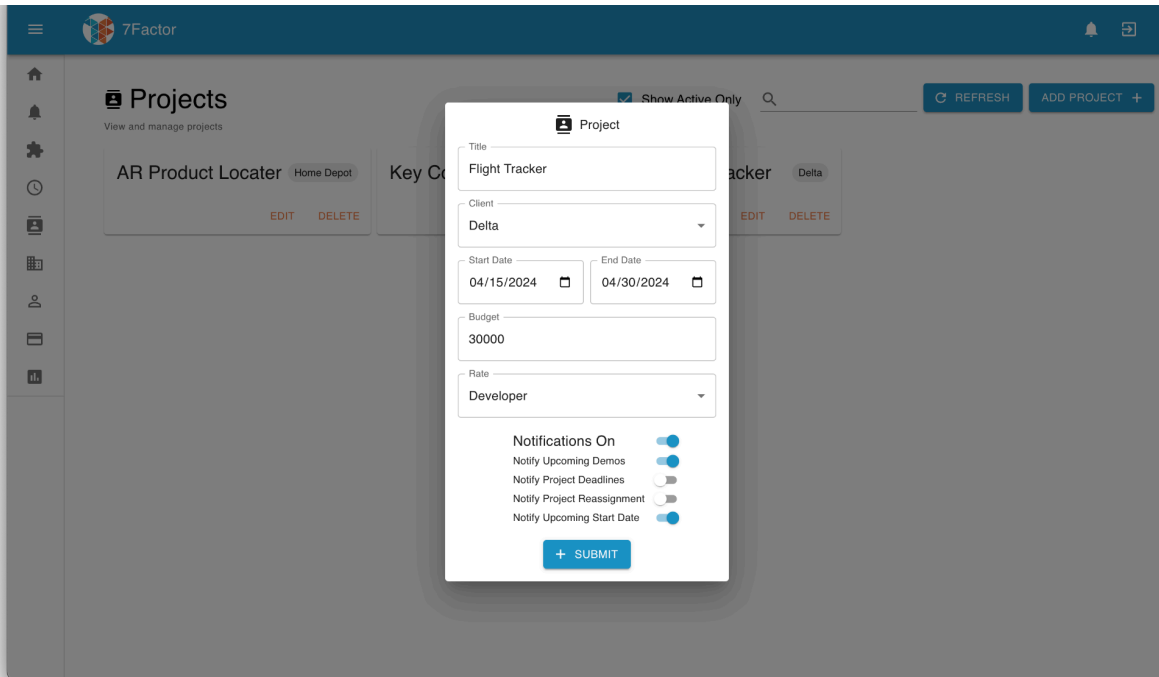
**Figure 4:** Notification settings can be enabled and disabled during project creation.

Once the project is created, the notification settings that were chosen will be set for that project. However, to add employees to a project, they must be added through the project management dialog or Skill Match Search. The previously mentioned notification settings can also be modified in the project management dialog (as seen in Figure 5).
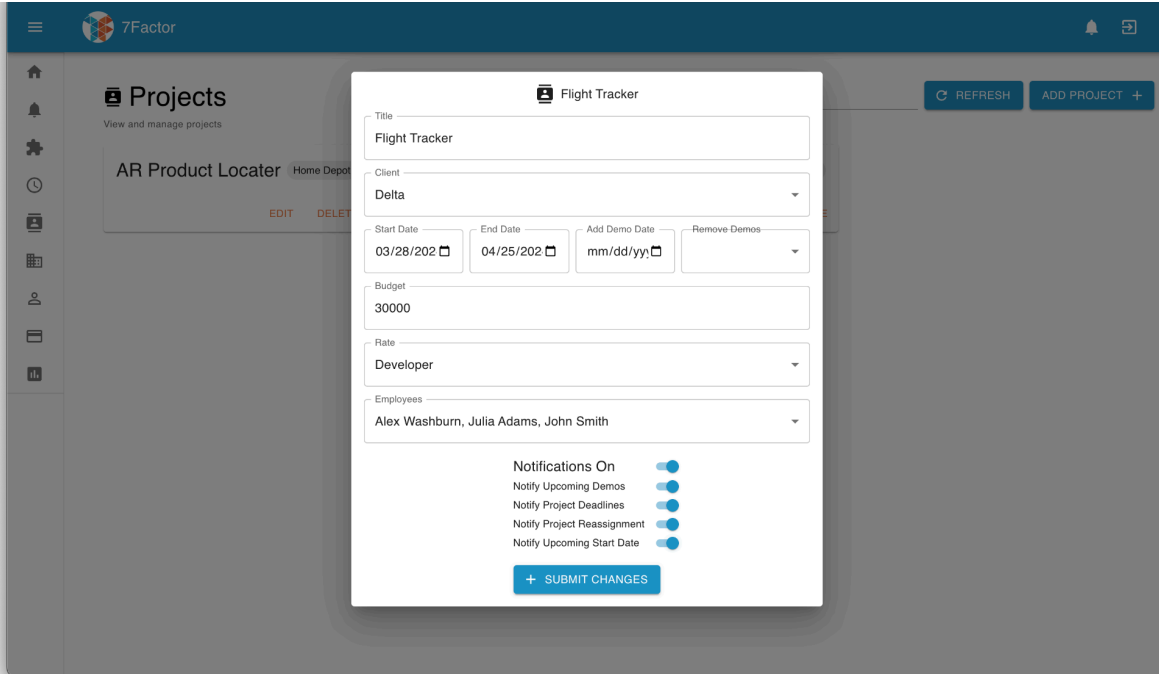
**Figure 5:** Notification settings can be enabled and disabled in the project management dialogue.

After the employees are added to a project this way, the specified notifications will either be sent out immediately or scheduled to be sent at a later date, depending on the type of notification. The types of notifications and the conditions for when they will be sent out to employees are specified in Table 1.

**Table 1:** *Notification Types and Conditions to be Sent to a User*

| Notification Type | Conditions |
|---|---|
| Start Date | Sent out immediately if an employee has been assigned within 2 weeks of a project's start date. |
| | Sent out 2 weeks before the start date of a project if an employee has been assigned before this period. |
| | Sent out on the Wednesday before the start date of a project if the start date is on a Monday. |
| Deadline | Sent out to an employee 2 weeks before the deadline. |
| | Sent out to an employee 1 day before the deadline. If the deadline is on a Monday, it is sent out on the previous Friday instead. |
| Demo | Sent out to an employee 3 days before a demo. |
| Reassignment | Sent out immediately to an employee who has been moved to an active project. The remaining employees on the first project won't be notified. |
| Timesheet Reminder | Sent out on Fridays to all employees. |

If a project is deleted, all of its notifications will also be deleted by removing them from the notification database.When a notification type toggle is turned off in project management, reassignment notifications will be archived, and all other notifications will be deleted. If an employee is removed from a project, notifications from that project will also be removed from that employee's notification inbox (as seen in Figure 6).
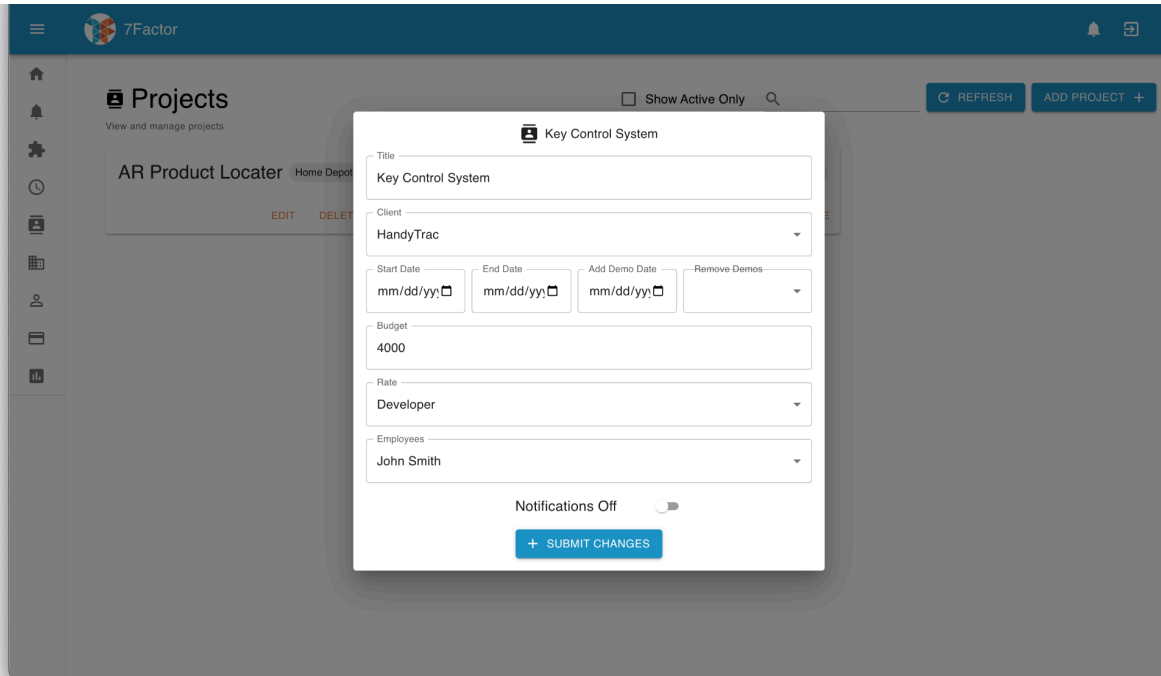
**Figure 6:** Employee removal and project deletion will result in notification removal.

## Employee Usage

Employees will be able to view the Notification Page by clicking on the bell icon located in the Taskbar or Menu bar. Notifications that the employee has received will appear in the inbox, with the most recently received notifications appearing at the top of the list. Each notification will display the following information: title, message, type, client name, project name, and send date. Each notification — other than timesheet notifications — will display the following information: title, message, type, client name, project name, and send date (as seen in Figure 7).
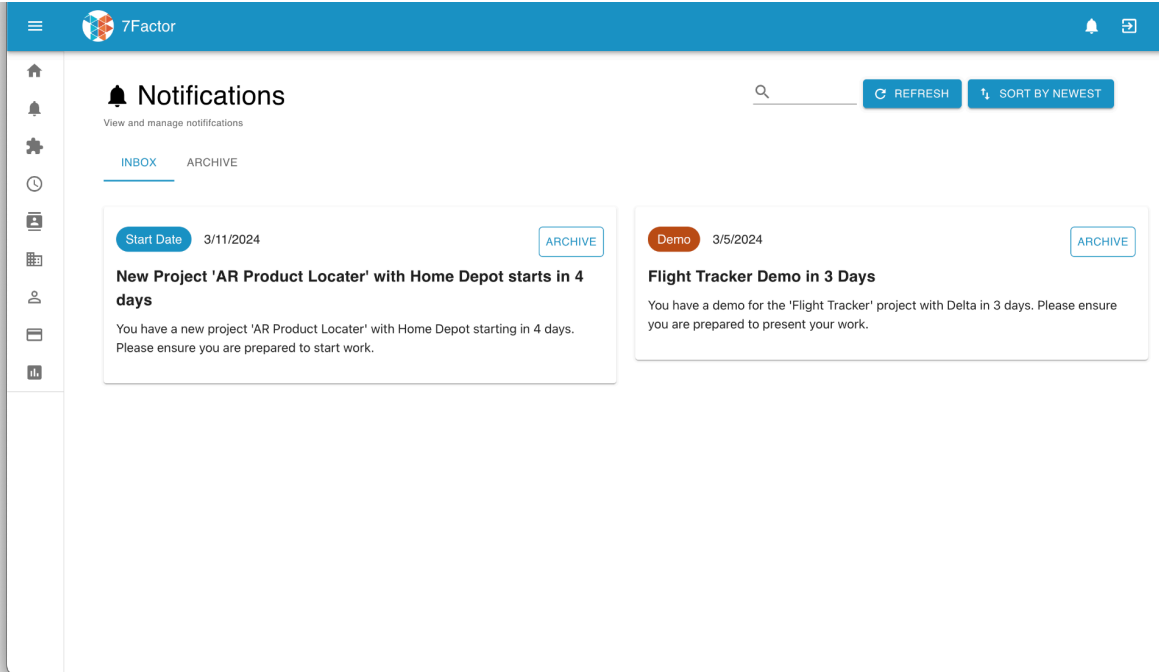
**Figure 7:** *Notification Page:* An employee's notifications are displayed on the notification page.

The search bar at the top of the page can be used to find notifications quickly. Search queries that match a notification's title, message, type, client name, or project name will display those matching notifications (as seen in Figure 8).
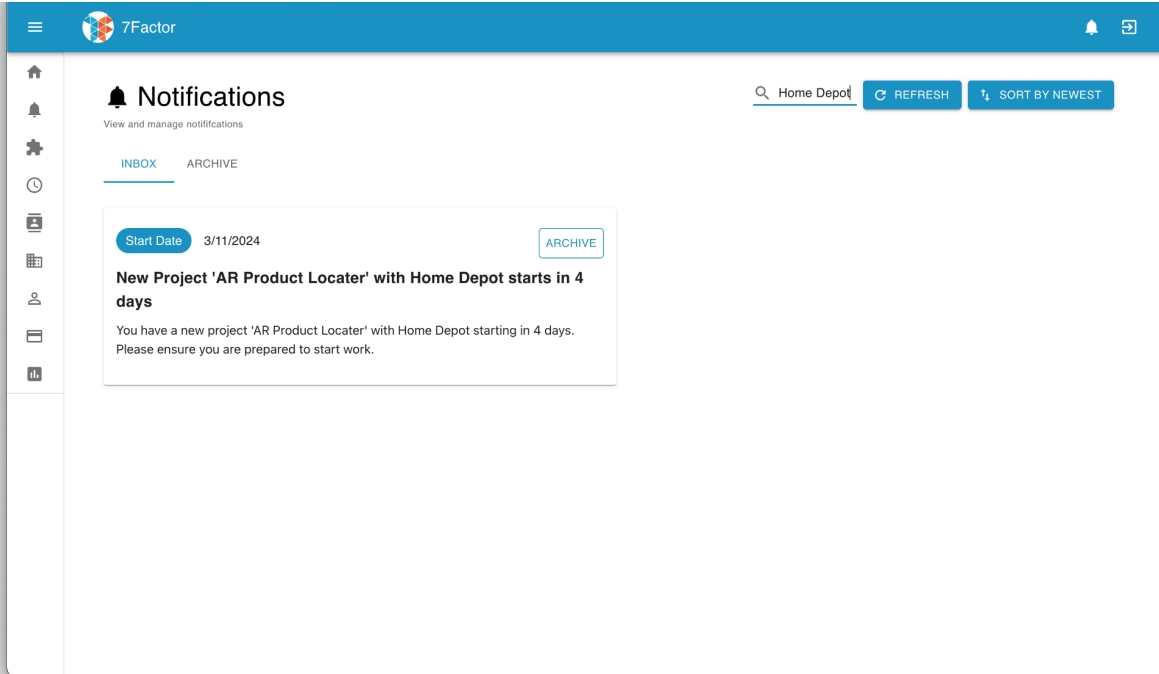
**Figure 8:** Notifications are searchable by their contents.

To ensure the most recent notifications have been received, the refresh button can be clicked to display the most up-to-date version of a user's inbox (as seen in Figure 9).
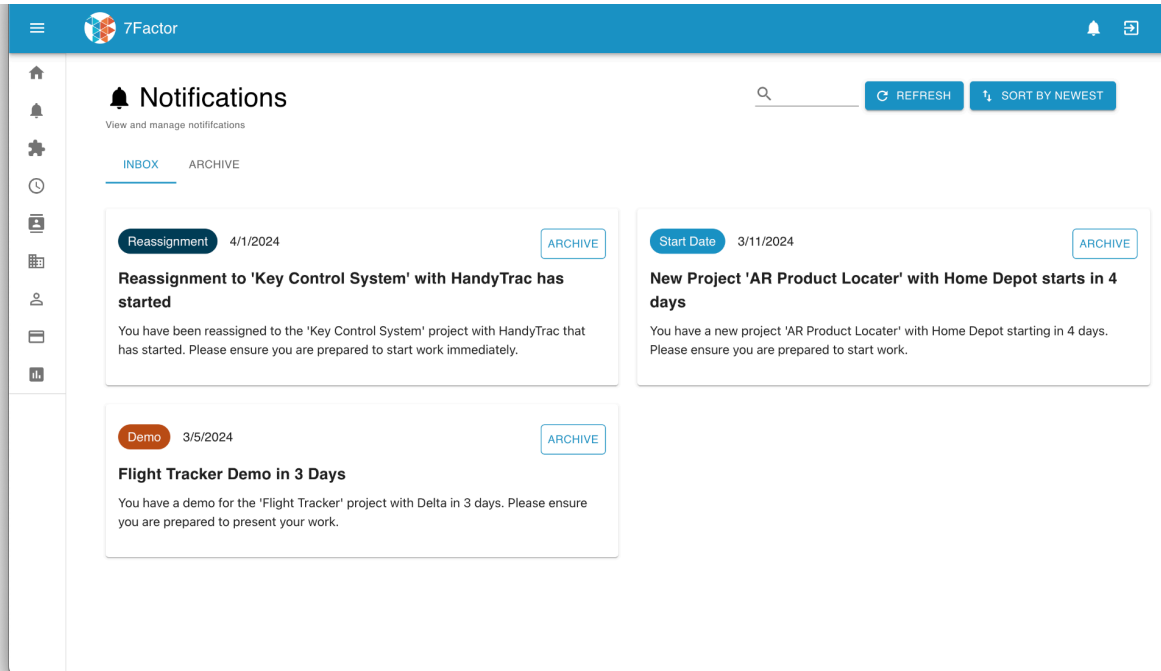
**Figure 9:** The notification page can be refreshed to stay up-to-date with the latest notifications.

Once a project has been completed, the notifications attributed to it will automatically expire and be sent to an archived folder. Notifications can also be sent to the archive folder manually by clicking the archive button on the desired notification. From the archive folder, notifications can be restored to the regular inbox or permanently deleted (as seen in Figure 10).
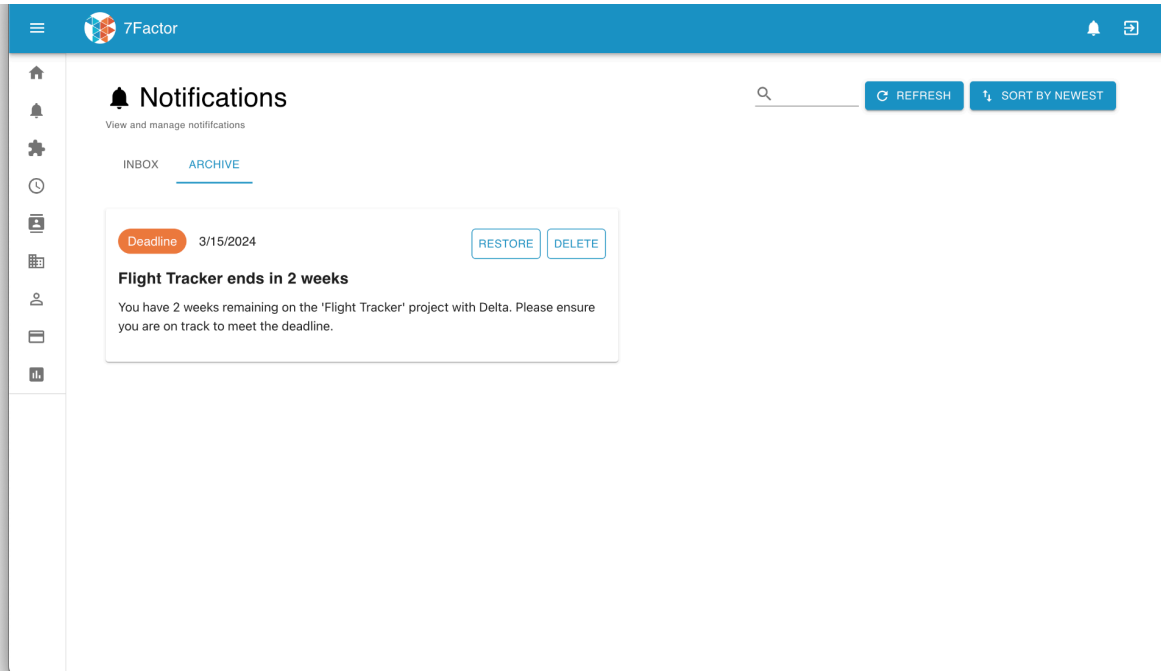
**Figure 10:** *Notification Archive*: Notifications can be archived, restored, and deleted permanently.

## Skill-Match Search

The following section outlines the additional functionality brought to the Staffing Tool with the Skill-Match Search Page. The process by which an administrator would search for employee skills and assign desired employees to a project will be shown.

Administrators will be able to view the Skill-Match Search Page by clicking on the puzzle icon located in the Taskbar or Menu bar. A list of all employees will be visible, with each employee having a card. Each employee card displays an employee's name, ID, skills, and a button that can quickly add or remove them from a project of the administrator's choosing (as seen in Figure 11).
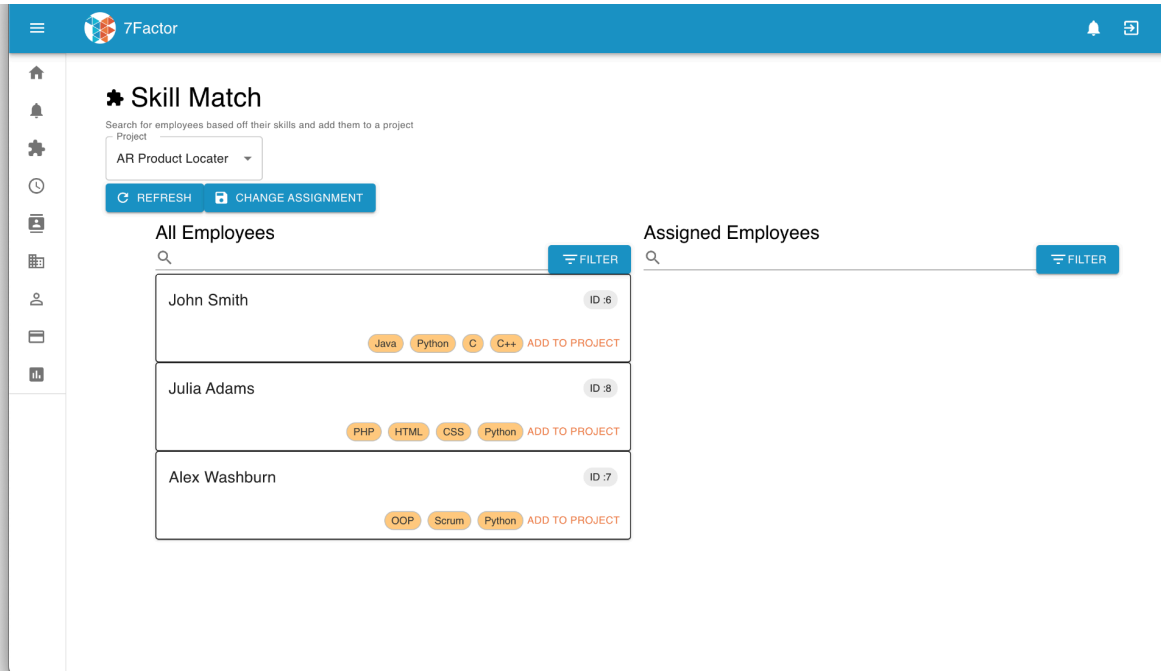
**Figure 11:** *Skill-Match Page:* Employees are displayed on the skill-match page with their relevant skills.

Employees will be displayed if a search query matches their name or a skill that they possess. Similarly, employees can be filtered according to their skills using the filter button (as seen in Figure 12).
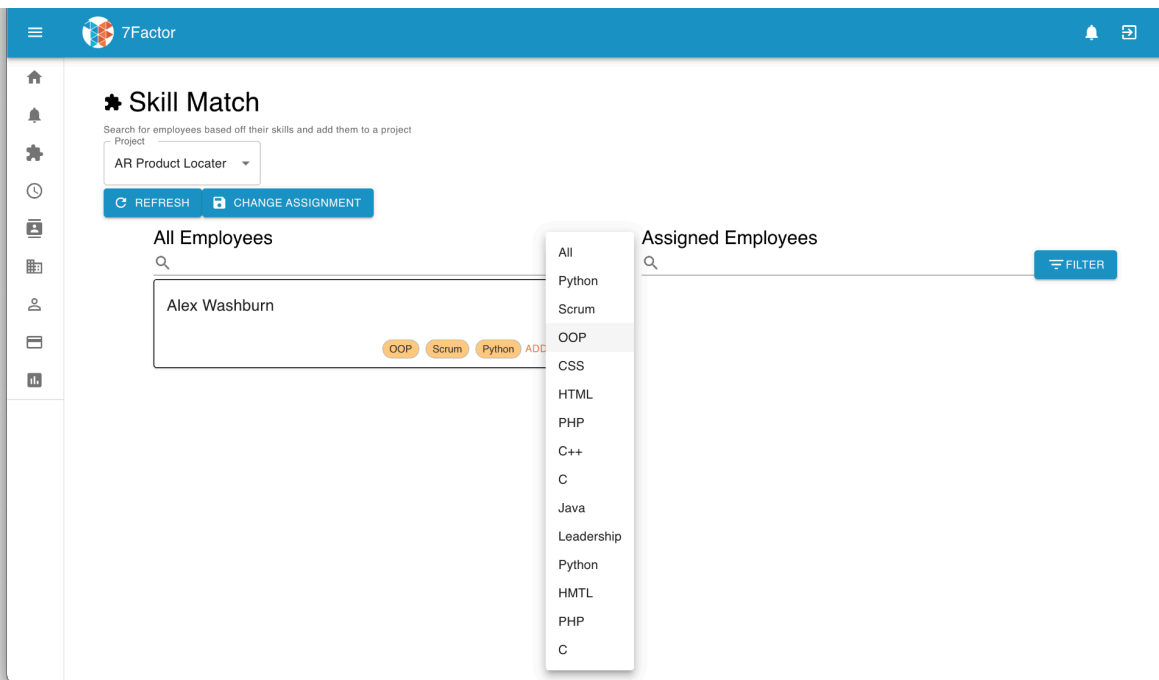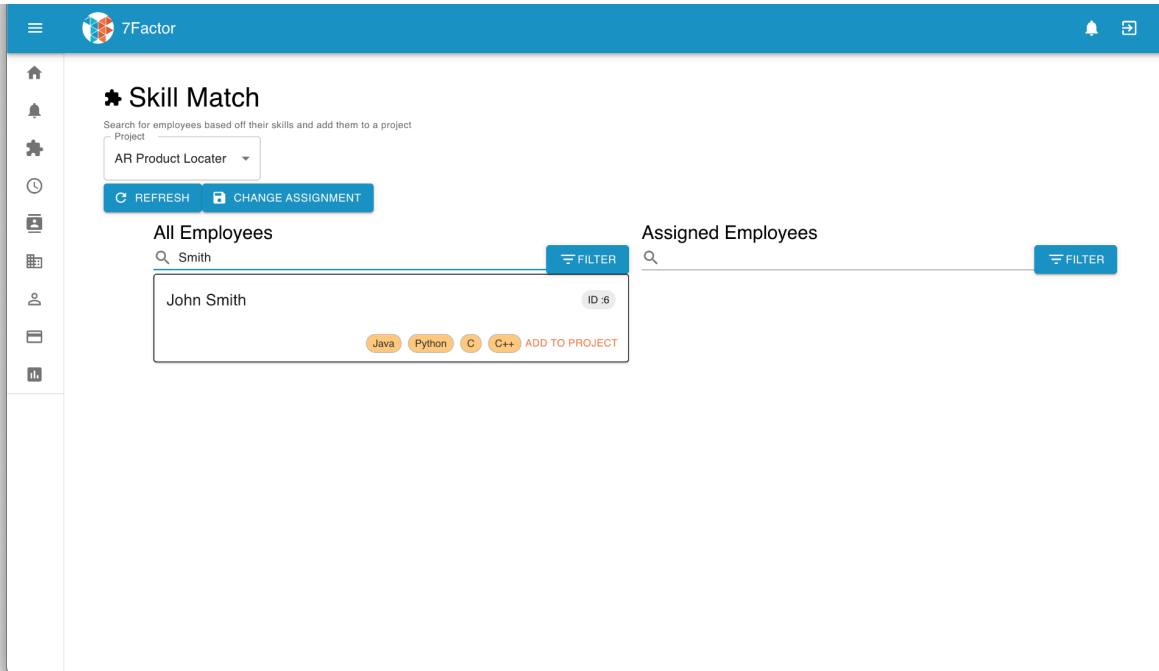
**Figure 12:** *Notification Search & Filter:* Employees are searchable and filterable by their names and skills.

Once a desired employee is found, they can be added to a project by clicking the corresponding button on the employee's card. A dialog will be produced, where an administrator can choose which project to assign the employee to. Adding an employee this way will also ensure that the project's notifications are automatically sent to them. Upon removal of an employee from a project, all notifications related to that project are deleted from the employee's notification inbox.
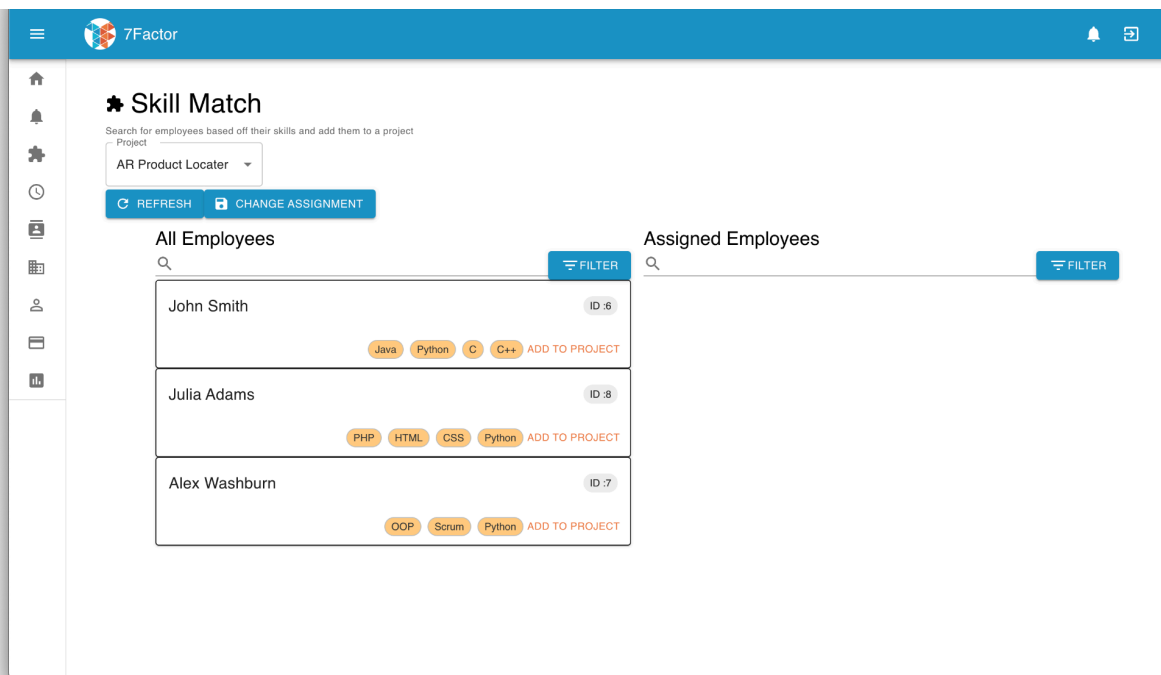


**Figure 13:** *Skill-Match Employee Management:* Employees' assignments can be altered using skill-match.

# Recommendations and Future Work

## Advanced Analytics Features

Incorporating advanced analytics and reporting features would empower 7Factor's management with insights into project performance, employee productivity, and resource allocation. Customizable dashboards and real-time analytics could facilitate strategic decision-making and operational optimizations.

## User Customization Options

Allowing individual users to customize notifications and dashboard views according to their preferences and job roles could enhance user engagement with the tool. Personalization options could improve the overall user experience, making the tool more intuitive and effective for daily use.

## Security Enhancements

This is a recommendation left by the previous MQP group. As the tool becomes increasingly integral to 7Factor's operations, ensuring high levels of data security and privacy compliance should be a priority. Future developments should focus on enhancing encryption methods, securing data storage, and implementing robust access controls.

## Data Transfer Objects

This is also a recommendation left by the previous MQP group. We recommend changing the way that data is sent between the front end and back end. The application currently sends the entire server-side object representation regardless of what type of object is being requested. The implementation of data transfer objects would make the application much more efficient as only the relevant and necessary data would be transferred.

# Conclusion

In conclusion, the enhancements introduced to the 7Factor Staffing Tool represent a significant stride toward refining the company's operational efficiency and project management capabilities. The integration of an automatic notification system within the Staffing Tool has addressed critical gaps in time management and deadline tracking, fostering a more organized and efficient work environment for 7Factor's globally dispersed team. This project not only alleviates the administrative burden associated with manual deadline reminders but also underscores the potential for custom-developed solutions to surpass the functionalities of off-the-shelf software like TimeIQ. Through a meticulous design and implementation process, we have demonstrated the feasibility and effectiveness of tailoring software solutions to meet the unique needs of a company, thereby enhancing its ability to manage projects and monitor employee productivity more effectively.

# Appendices

## Development Environment Setup

You must install all the following tools (these are the versions that were being used) before running the Staffing Tool Platform:

- Docker (version 4.12.0)
- Git (version 2.39.2)
- Java (version 8, jdk "1.8.0_341")
- NodeJS (version v16.17.0)
- NPM (version 8.15.0)

If you are running the Ubuntu Linux operating system, you can run this command in a terminal to install all the packages at the same time:

```
sudo apt-get install docker git nodejs npm
```

## Running the Staffing Tool Platform Code

If you run into any difficulties getting the staffing tool to run, please do feel free to email our group at the emails on the Digital WPI page. Our team and previous teams did run into trouble trying to initially run this code and we are willing to help future teams get started. Additionally, 7Factor has been very helpful to our group and can be a valuable resource for you if you encounter any issues. A common issue included errors while setting up docker containers for the database and the API, which were resolved by altering local environment files to be compatible with your device.

To run the codebase, you must first get access to the GitHub repository that contains the source code for the front end, back end, and database. To start your project, you will fork 7Factor's

Staffing Tool repository once you're given access to it. This fork is where all of your project development will occur. At the end of your project, you will submit a pull request against the repository to finalize your changes.

Once you've forked and cloned the repository, you should first run the database, then the back end API, and finally the front end.

## Database

- Open the database repo folder in a terminal (bash works, prompt does not)
- Run the following command
    - `./env/_up.sh`

## Back End API

- Open the back end repo folder in a terminal (bash works, prompt does not)
- Run the following command depending on your operating system
    - Windows: `./gradlew.bat bootRun`
    - Other: `./env/_up.sh`

## Front End

- Open the front end repo folder in a terminal (bash works, prompt does not)
- Run the following command
    - CD into the src folder: `cd src`
    - `npm install`
    - `npm run start`

# References

Banker, K. (2020). Web Development with MongoDB and Node.js. Packt Publishing.

Cohen, D., Lindvall, M., & Costa, P. (2004). An Introduction to Agile Methods. Advances in
 Computers.

Farrell, J. (2015). Programming Logic and Design, Comprehensive. Cengage Learning.

Fitzgerald, B. (2017). The Art of Project Management for Software and IT Engineers. Springer.

Flanagan, D. (2018). Java in a Nutshell: A Desktop Quick Reference. O'Reilly Media.

Gaikwad, S. (2018). Learning React: A Hands-On Guide to Building Web Applications Using
 React and Redux. Pearson Education.

Heckler, M. (2021). *Spring Boot Up & Running*. O'Reilly Media.

Kane, S. P., & Matthias, K. (2018). *Docker: Up & Running, 2nd Edition*. O'Reilly Media.

Kerzner, H. (2017). Project Management: A Systems Approach to Planning, Scheduling, and
 Controlling. John Wiley & Sons.

Lock, D. (2014). The Essentials of Project Management. Gower Publishing, Ltd.

McFarland, D. (2021). JavaScript & jQuery: The Missing Manual. O'Reilly Media.

Obe, R. O., & Hsu, L. S. (2017). *PostgreSQL: Up and running, 3rd Edition*. O'Reilly Media, Inc.

Pavithra, K. (2020). Mastering Material-UI: Build Modern React Applications Using
 Material-UI. Packt Publishing.

Steinmetz, N., & Wattenberg, F. (2019). React Material-UI Cookbook: Build captivating user
 experiences using React and Material-UI. Packt Publishing.

Thoring, K., Müller, R. M., & Desmet, P. (2021). Figma: An Emerging Cloud-Based Tool for
 Collaborative Interface Design. In Proceedings of the Design Society.

Waloszek, G. (2018). User Experience Design: Creating Designs Users Really Love. Apress.