April 2019

Data Manipulation and Visualization in the Washburn Shops Lab Management System

Samuel Baumgarten
Worcester Polytechnic Institute

Follow this and additional works at: https://digitalcommons.wpi.edu/iqp-all

Repository Citation

This Unrestricted is brought to you for free and open access by the Interactive Qualifying Projects at Digital WPI. It has been accepted for inclusion in Interactive Qualifying Projects (All Years) by an authorized administrator of Digital WPI. For more information, please contact digitalwpi@wpi.edu.
Data Manipulation and Visualization

in the

Washburn Shops Lab Management System

An Interactive Qualifying Project Report

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the Degree of Bachelor of Science

in Computer Science

by

___________________________________
Samuel M. Baumgarten

April 25, 2019

Professor Michael J. Ciaraldi, Major Advisor
The Washburn Lab Management System was deployed in January of 2019 to collect more information about the usage of the manufacturing laboratories at WPI. This project’s primary objective was to build tools to understand the data generated by the system and to provide an interface for the manipulation of stored information. In pursuit of these goals, we conducted an analysis of multiple business intelligence (BI) tools, created example reports in the selected BI tools for use by the facility staff, and developed an administration interface designed for day-to-day information manipulation. The tools developed by this project have been successfully deployed and utilized to make data-driven policy decisions that have an influence on the operations of the manufacturing laboratories at WPI.
# Table of Contents

Abstract ........................................................................................................................................... i

1 Introduction ...................................................................................................................................... 1
  1.1 Background ............................................................................................................................ 1
  1.2 Operational Usage ............................................................................................................. 1
  1.3 Data Comprehension ...................................................................................................... 2
  1.4 Data Manipulation .......................................................................................................... 2

2 Data Comprehension .............................................................................................................. 3
  2.1 Goals ....................................................................................................................................... 3
  2.2 Platform Selection .......................................................................................................... 3
    2.2.1 Evaluation ............................................................................................................... 4
    2.2.2 Selection ............................................................................................................... 12
  2.3 Data Connectors ........................................................................................................... 12
  2.4 Example Visualizations and Dashboards ...................................................................... 15
    2.4.1 GDS Reports .......................................................................................................... 15
    2.4.2 Power BI Reports .................................................................................................. 17
  2.5 Access ................................................................................................................................... 18

3 Data Manipulation ................................................................................................................ 19
  3.1 Overview ....................................................................................................................... 19
  3.2 Access ........................................................................................................................... 20
  3.3 Interface Pages ............................................................................................................. 20
    3.3.1 Devices .................................................................................................................. 20
    3.3.2 Lab Users .............................................................................................................. 21
    3.3.3 Quizzes and Quiz Groups ...................................................................................... 24
    3.3.4 Shops .................................................................................................................... 26
    3.3.5 Shop Authorizations ............................................................................................. 27
    3.3.6 Tools ..................................................................................................................... 28
    3.3.7 Tool Authorizations .............................................................................................. 28
  3.4 Interactive Console ....................................................................................................... 29
    3.4.1 Console Access ...................................................................................................... 29
3.4.2 Query Interface.....................................................................................................31
3.4.3 Tasks from Console...............................................................................................32
4 Conclusion and Future Work ..........................................................................................33
1 Introduction

1.1 Background
The WPI Mechanical Engineering Department operates two manufacturing labs on campus – the Washburn Shops and Higgins Labs. An estimated 1,600 students register to use these shops each year to make use of the mills, lathes, and other various tools that members of the WPI community can utilize. In 2018, an effort was started to develop a system to improve lab safety, increase access to the facility and better understand how users made use of the facilities. This system, known as the Washburn Lab Management System, was developed as an MQP and deployed in January of 2019.¹

1.2 Operational Usage
The introduction of the lab management system has changed the way the manufacturing facilities handle new user registration, usage tracking, and authorization management.

When a new user wants to gain access to a manufacturing facility at WPI, he can visit the manufacturing labs website at https://mfelabs.org. On this page, he will find a list of quizzes referred to as the "Basic User Training." To obtain permission to use WPI's manufacturing resources, a user needs to complete all the quizzes in the "Basic User Training" module. Upon completion of these quizzes, a user is granted access to the facility as a "basic user" meaning he can utilize the space during its regular operating hours as well as at any time a lab monitor opens the facility.

The first time a user attempts to check-in to a WPI manufacturing facility, the entryway kiosk will prompt him with a brief registration wizard that will allow him to associate a WPI ID card with his account, confirm his name, and take a photo.

Each time a user enters or exits a manufacturing lab, he is required to tap his WPI ID badge on the check-in or check-out kiosk situated at the entrance to the lab (See Figure 1). When a user checks-in, he will either receive a success message or an error message if he is not authorized to enter the facility. If a user successfully checks in, a badge listing the user's name and the tools he is allowed to use will print. At this point, the lab monitor will receive a notification on the controller device stating that a user has entered the facility.

1.3 Data Comprehension
The MQP that developed the Washburn Lab Management System focused on the development of the system’s core functionality and data collection mechanisms. One of the core goals of this project was to utilize the data generated by the pre-existing system to provide actionable insights to relevant stakeholders. To this end, we developed data visualizations that present current and historical data in an easy to comprehend manner (See Figure 2).

We conducted a thorough analysis of various business intelligence and data analysis tools to select tools that fulfilled the key requirements identified in the planning stage of this project.

1.4 Data Manipulation
The lab management system dramatically reduces the amount of manual data entry needed by the shop staff through the use of self-service portals and quizzes; however, it is still necessary for the shop staff to have a convenient way to enter and modify data in the system. One of the goals of the project was the development of an administrative panel that provides low-level access to the data stored by the system. This administrative panel enables shop staff to solve common problems such as lost IDs or incorrectly entered names and is where they can change user permissions, grant permissions, and approve pending authorizations for tools that require a manual sign-off.
2 Data Comprehension
The Washburn Shops Lab Management System generates data that is useful in analyzing the usage patterns of the manufacturing labs at WPI. We identified goals and use-cases for a system that visualizes this data, selected a platform that enables the fulfillment of these goals, and built example visualizations using the platform that can provide actionable insights to the shop staff. This section will detail the selection process of the visualization platform, provide examples of reports/visualizations we developed, and document how the visualization platform integrates with the primary lab management system.

2.1 Goals
We began the process of deploying a data comprehension tool for the Washburn Shops Lab Management system by identifying key goals that we intended for the data comprehension system to fulfill. We met with the Washburn shops staff to gain a better understanding of what information they thought would assist them in managing the manufacturing facilities.

Through these conversations, we derived a list of three goals we hoped to achieve by deploying a data comprehension tool:

- Better understand facility usage.
- Provide insights to enable more efficient scheduling of help hours.
- Enable the review of high-level shop event logs after an incident has occurred.

2.2 Platform Selection
Business intelligence (BI) platforms are tools that are designed to provide business leaders access to insights that can help them in making informed decisions. These tools are designed to make it simple to analyze and summarize data to identify trends, identify strategic opportunities, and measure the impact of prior decisions. There are many market leaders in the BI space with each company's product having a unique set of strengths and weaknesses. One of the primary limitations we faced in the selection of a BI tool was the significant price of many of these platforms. BI platforms are becoming an essential tool for any enterprise. A 2010 study found that BI has been one of the top three agenda items identified by senior executives from 2003-2010.² The immense demand and tremendous value BI tools can deliver to companies have led many BI platforms to focus on large enterprise clients. This focus has allowed these companies to set their prices such that they are not realistic for many organizations that are unable to justify the expense with sales as is the case for this platform.

During the platform selection process, we evaluated five options:

- Custom developed solution
- Google Data Studio

• Microsoft Power BI
• Amazon QuickSight
• Tableau

Our evaluation methodology was based on a 2005 paper on the cost of commercial off-the-shelf (COTs) software which outlined an approach to evaluating the risk of integrating COTs packages. The approach builds a profile of each package by assigning a value 1-10 relative to the other packages being evaluated to each of the following attributes:

- **Functionality**
  - “The extent to which the COTS package contains all the features and functions that the organization had formally identified for a candidate COTS solution.”

- **Reliability**
  - “The software's overall quality and stability.”

- **Cost**
  - “The total software acquisition and implementation cost.”

- **Ease of Customization**
  - “The extent to which the package could be easily modified to meet the responding organization's unique needs.”

- **Ease of Use**
  - “The intuitiveness of the user interface.”

The paper randomly surveyed 126 mid to senior level managers and identified the relative importance of each of the five factors that were found to be statistically significant. The authors used these relative importance values to compute a score for a given software package given its profile. The scores can be computed using the worksheet in Table 1.

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Rating</th>
<th>Weight</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionality</td>
<td>×</td>
<td>3.5</td>
<td>=</td>
</tr>
<tr>
<td>Reliability</td>
<td>×</td>
<td>3.3</td>
<td>=</td>
</tr>
<tr>
<td>Cost (1=high, 10=low)</td>
<td>×</td>
<td>1.6</td>
<td>=</td>
</tr>
<tr>
<td>Ease of Customization</td>
<td>×</td>
<td>0.9</td>
<td>=</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>×</td>
<td>0.7</td>
<td>=</td>
</tr>
<tr>
<td>Overall Score</td>
<td></td>
<td></td>
<td>=</td>
</tr>
</tbody>
</table>

*Table 1. Evaluation worksheet for computing the weighted score a COTs profile.*

### 2.2.1 Evaluation
#### 2.2.1.1 Custom Developed Solution
One of the first options we evaluated was to build a simple analytics panel into the shop management system itself. This panel would have been accessed through the administration

---

interface. The visualizations would have been built using frameworks such as D3. The benefits of building a custom tool are as follows:

- Enables building complicated visualizations that are uniquely relevant to our use-case.
- Users only have to visit one interface for administration and data comprehension.
- Removes all limitations that could be imposed by a third-party platform.
- No ongoing subscription cost.

The drawbacks of building a custom tool are as follows:

- Challenging to adapt to new visualization requirements due to the domain-knowledge in web applications needed to modify reports.
- Time-consuming to implement.

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Rating</th>
<th>Weight</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionality</td>
<td>8</td>
<td>3.5</td>
<td>28</td>
</tr>
<tr>
<td>Reliability</td>
<td>6</td>
<td>3.3</td>
<td>19.8</td>
</tr>
<tr>
<td>Cost (1=high, 10=low)</td>
<td>2</td>
<td>1.6</td>
<td>3.2</td>
</tr>
<tr>
<td>Ease of Customization</td>
<td>3</td>
<td>0.9</td>
<td>2.7</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>9</td>
<td>0.7</td>
<td>6.3</td>
</tr>
<tr>
<td>Overall Score</td>
<td></td>
<td></td>
<td>60</td>
</tr>
</tbody>
</table>

**Functionality**  
We assigned a score of 10 because we would be able to build the tool to provide the exact functionality needed without needing to work around any platform limitations.

**Reliability**  
We assigned a score of 6 because while the system would likely be initially reliable; however, no support or online resource would be available should the system malfunction in the future.

**Cost**  
We assigned a score of 2 because of the high labor cost involved with creating and maintaining the tool.

**Ease of Customization**  
We assigned a score of 3 because the tool is extremely flexible but difficult to change. Unlike COTs solutions, a custom platform is virtually infinitely customizable as it only needs to meet the needs of one customer; however, these customizations cannot be easily made by people without a programming background.
Ease of Use
We assigned a score of 9 because the tool would be purpose-built for this use-case. We would be able to simplify the user interface only showing relevant data and user interface components. COTS solutions would likely be unable to simplify their interfaces to the same extent as they need to be flexible enough to work for multiple use-cases.

2.2.1.2 Google Data Studio
Google Data Studio (GDS) is Google’s data visualization and business intelligence tool. The product is part of the Google Marketing Platform. It was taken out of beta and made generally available in September of 2018. GDS enables us to create data connectors which can then be used by people without a computer science background to visualize data with standardized visualizations such as graphs, tables, and scorecards. GDS also enables simple transformations such as inner joins to be performed on data from the data connectors.

The primary benefits of GDS are as follows:

- Enables shop staff to modify and create visualizations using a visual interface.
- Provides multiple standardized visualizations that can be used to preview data quickly.
- No subscription cost.

The drawbacks of GDS are as follows:

- Some visualizations are challenging to create due to limitations of the platform.
- Does not have strong support for access-level restrictions nor does it integrate with the existing shop single-sign-on (SSO) system.

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Rating</th>
<th>Weight</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionality</td>
<td>8</td>
<td>3.5</td>
<td>28</td>
</tr>
<tr>
<td>Reliability</td>
<td>7</td>
<td>3.3</td>
<td>23.1</td>
</tr>
<tr>
<td>Cost (1=high, 10=low)</td>
<td>10</td>
<td>1.6</td>
<td>16</td>
</tr>
<tr>
<td>Ease of Customization</td>
<td>8</td>
<td>0.9</td>
<td>7.2</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>8</td>
<td>0.7</td>
<td>5.6</td>
</tr>
<tr>
<td>Overall Score</td>
<td></td>
<td></td>
<td>79.9</td>
</tr>
</tbody>
</table>

Functionality
We assigned a score of 8 because GDS can fulfill a majority of the visualization requirements for the system. GDS fulfills two of the three goals outlined in section 2.1 but has a few limitations that make it unideal for fulfilling the goal of enabling the review of high-level shop event logs. One such limitation is its lack of support for sorting with minute or second level precision.

Reliability
We assigned a score of 7 due to the lack of direct enterprise-level support and the moderate level of quality of the GDS platform. GDS has a sizable user-base meaning that there are
multiple product forums to find solutions to problems. These forums make solving problems possible; however, it does not provide the same level of service as a typical enterprise software package would provide. While developing example reports, we occasionally confronted bugs in the software that interfered with the report authoring process.

Cost
We assigned a score of 10 because GDS is a free offering from Google.

Ease of Customization
We assigned a score of 8 because GDS makes it extremely easy to customize reports within certain bounds. Assuming future use-cases can make use of Google’s standard visualizations, people without expert-knowledge can customize the reports. Google recently announced the developer preview of a new feature called community visualizations which enables the development of some custom visualizations; however, these will likely require someone with a background in computer science to implement.4

Ease of Use
We assigned a score of 8 because GDS is a well-designed web application and the process of viewing reports is simple; however, there is limited documentation of the authoring process available, and Google does not provide a mechanism to integrate the application into the Washburn system’s administration panel. To access and author reports, users must maintain a Google account separately from their lab management SSO account and access the application from https://datastudio.google.com.

2.2.1.3 Microsoft Power BI
Microsoft Power BI is Microsoft’s offering in the business intelligence space. It comes in three packages: Desktop, Pro, and Premium. Premium is the offering designed for enterprise usage and allows for on-premise deployment. Pro is similar to Premium is only offered as a cloud and desktop service. The Desktop version is free; however, it does not have any collaborative or cloud-based services and only allows access to data through the native Power BI Desktop app. Power BI was launched in 2010 as a project called Project Crescent bundled with Microsoft SQL Server Denali.5

The primary benefits of Power BI are as follows:

- WPI has an existing relationship with Microsoft and purchases other enterprise software from them.

• Enterprise-level support is available.
• Allows the creation of dashboard as well as email reports.

The downsides to using Power BI include the following:

• Ongoing service fee.
• Authoring tool requires users to have direct access to the database on their local machines.
• Requires Windows computer to author reports.

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Rating</th>
<th>Weight</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionality</td>
<td>10</td>
<td>3.5</td>
<td>35</td>
</tr>
<tr>
<td>Reliability</td>
<td>9</td>
<td>3.3</td>
<td>29.7</td>
</tr>
<tr>
<td>Cost (1=high, 10=low)</td>
<td>4</td>
<td>1.6</td>
<td>6.4</td>
</tr>
<tr>
<td>Ease of Customization</td>
<td>7</td>
<td>0.9</td>
<td>6.3</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>9</td>
<td>0.7</td>
<td>6.3</td>
</tr>
<tr>
<td>Overall Score</td>
<td></td>
<td></td>
<td>83.7</td>
</tr>
</tbody>
</table>

**Functionality**

We assigned a score of 10 because Microsoft Power BI can fulfill all the goals set for the data comprehension system. Power BI has a tool called custom visuals that would allow for the development of custom visualizations for reports should a future feature not be included with Power BI.

**Reliability**

We assigned a score of 9 due to Power BI’s support from Microsoft and its large number of customers. We are confident that Power BI would continue to operate without significant maintenance. We deducted 1 point from the rating because of the need to use a desktop tool to author the reports. This may pose a challenge in the future should the tool need to be installed on WPI owned computers.

**Cost**

We assigned a score of 4 because Microsoft Power BI costs $10 per user per month which is more expensive than some of the other options being considered. In addition, Microsoft could change the pricing at any time.

**Ease of Customization**

We assigned a score of 7 because Power BI requires some setup to customize reports but is extremely powerful once a user configures the software. Power BI reports are viewable in a web browser; however, they need to be authored using the Power BI desktop application. To author reports, users would need to download Power BI, configure the Postgres adapter on their computers, and add a security group rule on the database to allow connections from their IP addresses.
Ease of Use

We assigned a score of 9 because Power BI has a multitude of features that make it extremely simple to consume data from reports. One such feature is the ability to have reports emailed to users.\(^6\) This feature would make it such that shop staff could receive daily or weekly email updates with relevant information while still being able to use the web panel to find more detailed information. We subtracted 1 rating point due to Power BI’s lack of a web authoring interface.

2.2.1.4 Amazon QuickSight

Amazon QuickSight is a service offered as a part through Amazon Web Services (AWS). AWS QuickSight is similar to the other BI tools we considered; however, it has a primary focus on machine learning which they refer to as "ML Insights."\(^7\) AWS charges $18 per month per user (billed annually) for authoring and a maximum of $5 per month for readers. It is accessible through the AWS web console which is already heavily used by other components of this project. The primary benefits of AWS QuickSight are as follows:

- Keeps all user information in the same data center.
- Data is accessible from the AWS console.
- Web-based authoring tool.

The downsides of QuickSight include the following:

- High monthly cost.
- ML features are not directly relevant to the goals for the data comprehension system.
- No support for custom visualizations.

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Rating</th>
<th>Weight</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionality</td>
<td>6</td>
<td>3.5</td>
<td>21</td>
</tr>
<tr>
<td>Reliability</td>
<td>9</td>
<td>3.3</td>
<td>29.7</td>
</tr>
<tr>
<td>Cost (1=high, 10=low)</td>
<td>2</td>
<td>1.6</td>
<td>3.2</td>
</tr>
<tr>
<td>Ease of Customization</td>
<td>4</td>
<td>0.9</td>
<td>3.6</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>7</td>
<td>0.7</td>
<td>4.9</td>
</tr>
<tr>
<td>Overall Score</td>
<td></td>
<td></td>
<td>62.4</td>
</tr>
</tbody>
</table>


Functionality
We assigned AWS QuickSight a score of 6 in the functionality category because the focus of the product team seems to differ from the requirements of this system. The lack of expandability also means that the system may interfere with future strategic objectives.

Reliability
We assigned a score of 9 due to the enterprise level support provided by Amazon for AWS products. In our testing, we found the application to be extremely reliable and robust. Another benefit is that QuickSight’s integration with other AWS services used in this system such as SSO and AWS security groups mean that the changes to other parts of the system infrastructure would propagate to QuickSight automatically.

Cost
We assigned a score of 2 as QuickSight is one of the most expensive BI packages we evaluated.

Ease of Customization
We assigned QuickSight a score of 4 in customizability due to its lack of custom visualizations and the inflexibility of its authoring interface. During our evaluation, we struggled to use the interface and often ran into limitations of the platform.

Ease of Use
We rated QuickSight with a 7 in ease of use due to its simple report reading experience. While the authoring tool was unideal, consuming reports was very straightforward. Its integration with our existing authentication solution would allow users to log in to QuickSight through our SSO panel. The QuickSight report home page shows a list of reports and a corresponding thumbnail for each making it easy to quickly find information.

2.2.1.5 Tableau
Tableau is a leader in the business intelligence and data analysis fields. In 2008, Tableau awarded the CODiE award for “Best Business Intelligence or Knowledge Management Solution” by the Software & Information Association’s (SIIA). Their primary offering, Tableau Creator, includes Tableau Desktop, Tableau Prep, and a license to Tableau Server or Tableau Online. Tableau offers almost every feature included in the other packages evaluated. WPI’s Institutional Research Office holds a limited number of rotating seats to Tableau that they redistribute every six months. Without access to one of WPI’s licenses, Tableau would cost a minimum of $70 per month, billed annually, with potentially higher prices if multiple users needed to analyze the data or view reports.

---

### Characteristics

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Rating</th>
<th>Weight</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionality</td>
<td>9</td>
<td>3.5</td>
<td>31.5</td>
</tr>
<tr>
<td>Reliability</td>
<td>10</td>
<td>3.3</td>
<td>33</td>
</tr>
<tr>
<td>Cost (1=high, 10=low)</td>
<td>1</td>
<td>1.6</td>
<td>1.6</td>
</tr>
<tr>
<td>Ease of Customization</td>
<td>5</td>
<td>0.9</td>
<td>4.5</td>
</tr>
<tr>
<td>Ease of Use</td>
<td>8</td>
<td>0.7</td>
<td>5.6</td>
</tr>
<tr>
<td>Overall Score</td>
<td></td>
<td></td>
<td>76.2</td>
</tr>
</tbody>
</table>

**Functionality**

We assigned Tableau a score of 9 in the functionality category due to its expansive feature-set and widespread industry usage. It would be able to achieve all of the goals established for this project as well as being expandable enough to handle nearly any future change. We deducted 1 point due to its lack of individually focused features on our use-case.

**Reliability**

Tableau was assigned a score of 10 in reliability due to its historical record of successful deployments and the significant number of companies and organizations that use it in environments that are more critical than ours. Tableau is offered in two packages: server and hosted. The server package is an on-premise solution for the Tableau backend enabling all data to be kept on-site. The hosted edition is functionally equivalent to the on-premise edition; however, Tableau manages the underlying infrastructure. The hosted version would likely be more reliable in our use-case as we do not have a dedicated operations team.

**Cost**

We assigned Tableau a score of 1 in cost as the monthly cost of $70 per user is the highest of all the packages we evaluated.

**Ease of Customization**

We decided on a score of 5 for customizability because Tableau comes with a multitude of standard visualizations built-in that work for a majority of situations. These standard visualizations are relatively simple to modify allowing users to author new reports without support. Tableau is also extremely expandable, allowing for customized graphics. The drawback of Tableau's expansion system compared to other offerings we considered is that Tableau's expansions are challenging to develop. Developing a new visualization requires the user has a strong computer science background.

**Ease of Use**

We assigned Tableau an "ease of use" score of 8. The active Tableau community both at WPI and online is the reason for this score. Tableau is a moderately difficult software package to
learn to use; however, it is easy to find resources online. Reports generated by Tableau are simple to navigate which would be the primary use of the system.

2.2.2 Selection

After completing the analysis of the various BI software packages as described in Section 2.2.1 above, we reviewed the data collected to decide which platform we would use to build the data comprehension tool.

First, we decided to eliminate Tableau from the selection process due to its high price of $70 per month per authoring user. The additional benefits from Tableau would not have justified the excessive cost compared to the other packages considered.

We eventually shifted our primary focus to Google Data Studio and Microsoft Power BI, the two packages with the highest COTs evaluation scores (See Table 2). Both received high scores in all categories with the primary differentiator being the prices and feature-sets. Google Data Studio is a much less mature product than Power BI and doesn’t have nearly as extensive of a feature set; however, it can achieve the specific goals outlined in Section 2.2.1. Google Data Studio does not charge anything for the use of their service whereas Power BI costs $10 per user per month.

Ultimately, we chose Google Data Studio (GDS) to be the primary data comprehension tool we would work with due to its inclusion of our essential features and its lower cost. We did identify that Power BI may be an option eventually should the data comprehension needs of the system be expanded. For this reason, we created data connectors and example reports for both GDS and Power BI to simplify any future migrations with a focus on GDS as it is what the shop staff will initially utilize for accessing the system data.

2.3 Data Connectors

Importing, aggregating, and transforming data from multiple sources are essential features of both Google Data Studio and Microsoft Power BI. Data importing is completed using a concept called data connectors. Data connectors are mechanisms that enable a user to “connect” a data source to a BI tool. Each of the two BI packages includes multiple built-in connectors. GDS includes integrated connectors to services such as Google Analytics, Google AdManager, Google Sheets, BigQuery, and YouTube analytics. Power BI includes connectors for services and applications including Excel and Sharepoint.

Both of these BI packages can also be configured to connect to database or data warehouses. There is a clear distinction between database and data warehouses in the context of business intelligence tools. While both databases and data warehouses store data, databases are often developed to support performant insert, delete, update, and select operations. Data
warehouses are designed to be used for analytics tasks where fast selection queries are more important than quick updates or deletes. Generally, systems use a regular database for their production web applications while periodically copying some of the data from the production database into a data warehouse which users can use to perform analytics queries.

In the case of this system, we opted not to use a data warehouse such as Google BigQuery or AWS Redshift due to the relatively small amount of data the system collects and the significant cost of data warehousing solutions. The lowest cost AWS Redshift instance costs $180 per month while providing limited value to the system. In large systems, data warehousing can become essential; however, we have found that with optimized indexes, we can successfully perform most analytics queries directly on our production database. If this eventually becomes a performance bottleneck, it can be temporarily rectified by creating read-replicas of the database to separate analytics queries from production queries with the eventual goal of shifting to a data warehouse.

The production database for the Washburn Lab Management System is stored in a PostgreSQL (Postgres) instance running on Amazon RDS. We configured a database user specifically for each data analysis tool. These users have their permissions set such that they can only select data (update, delete, and insert are restricted) and sensitive data such as RFID numbers or WPI ID numbers are not accessible. The column level restrictions are implemented using views. For example, the “lab_users” table was made inaccessible to analytics user, and a new view called “lab_user_analytics” was created that only allows access to permitted fields.

GDS refers to data connectors and data sources (See Figure 3). It only allows one table or query to be executed for each source. This means we needed to create multiple data sources that referenced the same server to enable users to access all the tables needed. This presented a challenge when too many users accessed the visualization tools simultaneously as each data source opens its own server connection. This can cause a significant load on the server leading to it preventing new clients from connecting. Another consideration with GDS’s data source capabilities is that it currently has limited support for aggregation. This means that when trying to visualize information such as the number of users in the shop at a given time, we needed to create a data source with a custom query that would run on the database server.

Power BI does not come with full Postgres support by default. The option for Postgres will show in the menu; however, users will receive an error message if they try to utilize it. To use the
connector, users must install npgsql on their computers from Github. Power BI will show a prompt instructing the user to do this if a .NET Postgres data provider is not installed. One other limitation of Power BI’s Postgres support is that it requires a user to have the ability to connect to the database directly. GCS does not have this requirement because it is cloud-based. This enabled us to only grant access to Google’s servers without having to authorize individual users. There are multiple ways to handle this. The least idea is manually adding security exceptions to our RDS security group in the AWS console whenever a user wants to connect. This is unideal because the access is tied to a specific IP address meaning if the user gets a new address, their access will cease to work, and an unauthorized user could attempt to access the database if they were assigned the old IP address. Another solution is to set up a VPN that users could connect to when they wanted to access the database. This approach does not have the downsides of the direct access model; however, it would require the maintenance of a VPN server. The final approach is the one we used in testing whereby users do not access Power BI directly from their computers. Instead, they initiate a remote desktop connection with a remote server running on EC2 that has the security clearance to access the database. An added benefit of this approach is that all of Power BI’s calculations run on the remote server instead of on the user’s computer. This removes the potential for issues caused by a user trying to use Power BI on an underpowered system. It also makes it such that users don’t need to configure the Postgres adapter themselves as it is preconfigured on the server.

Power BI has extremely powerful data modeling and aggregation tools meaning we did not need to use SQL queries to transform the data retrieved from the Postgres database. We created a data connector for each table needed in Power BI which is automated by a wizard when you first create a new Power BI data connector. Once the data connectors were imported, we configured the relational model between the various tables (See Figure 4). Each model is associated with the other tables it has relationships with, and its primary/foreign keys are selected. Once Power BI has this information, it can be used to create aggregations, measures, and new columns. For example, we added a measure on “Lab Users” called “Time in Shop” which will be set to the sum of the duration of all of a user’s “Shop Sessions” in a given time period.
2.4 Example Visualizations and Dashboards
We created demonstration data comprehension dashboards in both GDS and in Power BI to provide basic information to the shop staff and to provide examples that can be built upon in the future.

2.4.1 GDS Reports
In GDS, we produced the following four reports:

- Daily Report
- Check-Out Report
- Event Report
- Monthly Usage Report

2.4.1.1 Daily Report
The daily report is the most involved of the four. It is intended to show how information relevant to the current state of the shop can be displayed. The information it displays is as follows:

- The current number of users in the lab.
- The number of people that have visited the shop during the current day.
- The number of people that have completed the basic user training during the current day.
- The number of people that have registered during the current day.
- The number of users that haven’t checked out this in the current week.
- The permissions breakdown of people actively in the shop.
- A list of the lab monitors currently in the shop.

This information can be used to give a member of the shop staff a broad overview of what is occurring in the shop and any unusual trends such as an unusually high number of people coming to the shop.

2.4.1.2 Check-Out Report
The check-out report is intended to help shop staff track the effectiveness of policies that encourage people to check-out. When users leave the shop without tapping their badge on the check-out kiosk, it decreases the reliability of the information generated by the system. This report includes the following elements:
• A leaderboard of the users that forget to check-out most.
• A line graph showing the number of users that did not check-out by day.
• A metric showing the number of times a user did not check-out during the current week.
• A chart showing a breakdown of the permission levels of users that did not check-out.

This report can be used to understand who is not checking out and evaluate the effectiveness of changes that attempt to increase the number of users checking out.

2.4.1.3 Event Report
The event report is a straightforward report that solely includes a table listing "open_shop," "close_shop," and "become_lab_monitor" events. This report enables the shop staff to quickly check who the active lab monitor was at a specific time. Each event record includes the lab monitor’s name, the type of event, and the date the event occurred. “open_shop” events occur when a lab monitor checks into the shop when it is closed, and it automatically opens. “close_shop” events occur when a lab monitor uses the controller device to close the facility. “become_lab_monitor” events occur when the controller device is used to change the lab active monitor to a new person.

2.4.1.4 Monthly Usage Report
The monthly usage report provides insights regarding the number of users utilizing the facility in the past month and the ratio of basic users to lab monitors during this period. The report consists of a large graph with three data lines, one for lab monitors, one for basic users, and one for all users. The distance between the two lines indicates the ratio of lab monitors to basic users in the facility on that day. This chart can be used to explore the usage of the shop during various parts of an academic term and identify ideal maintenance windows.
2.4.2 Power BI Reports

In addition to the reports in GDS, we created reports in Power BI. We decided not to exactly replicate the GDS reports, instead focusing on increasing the breadth of our examples as it is not extremely difficult to copy the techniques used to create the reports between platforms.

The reports created in Power BI are as follows:

- Weekly Report
- Usage Report
- Badge Print Report

2.4.2.1 Weekly Report

The weekly report is a dashboard that provides the shop staff with everything they need to know about the weekly usage in a glance. It shows the number of times someone has come to the facility, the number of new users that have completed the basic user training, and the current pass rate for the quizzes in the basic user training. In addition, the report shows the trendline of the average quiz pass rate and the average pass rate by quiz. This helps identify areas of the training that need additional focus.

2.4.2.2 Usage Report

The usage report provides an example of a report that uses computed measures and can be used to identify the users that are spending the most time in the facility. It also introduces the concept of using conditional highlighting to bring attention to users that are violating shop policies. The usage table in the upper left corner of the screen includes the following information:

- The user’s name
- The average number of hours per day he has spent in the shop over the past seven days.
- The total time he has spent in the shop over the past seven days.
- The maximum number of contiguous hours he has spent in the shop over the past seven days.

Another unique feature of the usage report is its usage of Power BI’s drill-down functionality. We
utilized this featured such that a member of the shop staff can click on a user in the graph or table and view a list of the user’s shop sessions in the table on the right side of the dashboard (See Figure 10). This feature enables the shop staff to obtain detailed information about how a user has been using the facility if they see they have extremely high usage.

2.4.2.3 Badge Print Report

The badge print report is a report used to measure the impact of the badge printing policy deployed in March of 2019. As described in the MQP paper, we began requiring that all users in the shop wear a name badge that would automatically print when a user checked in. To understand the effectiveness of this policy in increasing check-ins, we created a report that showed the total number of check-ins, the number of check-ins per person per day, and a graph of the people with the most check-ins per day. This report was used in our evaluation of the policy and helped us discover that rolling out the badge-printing system increased the number of users checking-in per day by over 20%. We also found that more users checked-in multiple times per day with the average number of check-ins per person per day moving from 1.3 to 1.5.

2.5 Access

The GCS data visualizations can be accessed through the shared Google account that owns the visualization. To log in, use your Washburn SSO account to access the AWS console. Once in the AWS console, visit the AWS parameter store page. There you will find the google account credentials. Once you have those credentials, you can log in to the account at https://datastudio.google.com.

The Power BI visualizations are contained in a Power BI document that is stored in AWS CodeCommit. To access the visualization document, log in to the AWS console and visit the CodeCommit page. You can then access the “WashburnPowerBI” repository which contains the document. To view the visualizations using a cloud server, you can visit the EC2 service page and start the t2.large instance used for visualizations. If this instance doesn’t exist, it can be recreated from the snapshot in the “snapshots” tabs. Clicking the “Connect” button will provide instructions on how to connect. The password can be retrieved from the AWS Parameter Store service. Remember to shut down the instance after using it to prevent excessive charges.

Figure 11. A screenshot of the badge print report (data modified for illustrative purposes).
3 Data Manipulation

The Washburn Lab Management is designed to be a centralized repository for all the management information stored by Washburn Shops. The system autonomously handles processes such as basic user enrollment; however, other tasks such as data correction, permission assignment, and configuration need to be manually completed by shop staff. The administration panel is the primary interface shop staff use to manipulate data managed by the system. This section provides an overview of the administration panel and documentation describing how to complete everyday tasks. It also provides a basic overview of the command line interface intended for more advanced data aggregation and manipulation.

3.1 Overview

The administration panel is divided into two columns: the sidebar and the main content area (See Figure 12). The sidebar area shows the current user information and a list of various object types a user can manipulate. The administration panel is a low-level interface so most of these object types are directly related to database objects so an administrator can resolve most problems resulting from incorrect data being entered. The types of objects modifiable from the administration panel are as follows:

- Devices
- Lab Users
- Quiz Groups
- Quizzes
- Quiz Results
- Shops
- Shop Authorizations
- Shop Sessions
- Tools
- Tool Authorizations

A detailed overview of each of these object types can be found in section 4.4 of the Washburn Shops Information System MQP paper. In the following sections, this paper will detail some of the actions that can be performed on each of these object types.

Each page contains a list of all the objects stored in the system of the selected type. The header labels can be clicked to order the list by that field. Clicking a second time will switch the ordering direction. Clicking on a row will bring the administrator to a show page which provides

---

more detailed information about the object and will sometime show a list of related objects. Clicking the edit link on the list view or the edit button on the show view will bring the user to a form which can be used to modify a subset of the object’s fields (all fields can be modified through the command line console). Clicking the new button on the list page will enable an administrator to create a new instance of the selected object type.

As mentioned earlier, the administration panel is a low-level interface. The interface provides a significant amount of flexibility and control at the expense of enabling users to take invalid actions. Users should review the documentation regarding the usage of each model before making modifications not explicitly described in this document.

3.2 Access

Users can access the administration panel by visiting https://sms.mfelabs.org/admin. This URL will redirect users to the Washburn SSO sign-in page where they can authenticate using their credentials. Existing shop staff can add new users by visiting the AWS console, going to the SSO service page, adding a new user, and assigning them to a group with access to "Washburn Admin." When authenticating for the first time from a new device, users may be prompted to complete a two-step verification process in which Amazon will send the user an email with a code that he must enter after his password. If a user does not receive this email, he should verify it did not get placed in his junk folder. More information about two-step verification can be found at https://docs.aws.amazon.com/singlesignon/latest/userguide/enable-two-step-verification.html.

3.3 Interface Pages

3.3.1 Devices

A device is a client that can connect to the Washburn backend. Prior to a device being able to make API requests, it needs to be approved in the administration panel. When a device first connects to the backend, it will send a create device request. This will create a new entry in the “Devices” panel in the administration interface.

3.3.1.1 Approving Devices

To approve a new device, first, click on the device you wish to approve in the list of devices. Devices can be identified by their “Display Identifier.” Administrators should always ensure that the device identifier displayed on the new device matches the number shown in the administration panel. If there are a large number of devices in the system such that it is hard to identify the particular device, the search field in the upper-right corner can be used to search by display name. Once the device has been identified, press the edit button on the right side of the list (See Figure 13). This will bring the user to the edit device form. Here the user should select

Figure 13. A screenshot of the edit button’s location in the list view.
the shop this device is being deployed in, enter a display name to make it easy to identify the
device in the future, check the approved check box, and enter one of the roles listed below
based on the devices use. If the device is a check-in device and needs to print badges, the user
should enter the printer ID for the printer that badges should be printed to. Printer IDs can be
found by logging into the gr-lms@wpi.edu Google account and visiting
https://www.google.com/cloudprint. Clicking on the printer in the printers tab then clicking
show print jobs will show the printer ID in the search field in the format: “user:all
printerid:{printer_id_here}.”

All devices need to be assigned a role which will determine the level of access the device will be
granted. The valid options for the role field are as follows:

- “check_in”
  - Used for check-in kiosk devices.
- “check_out”
  - Used for check-out kiosk devices.
- “lab_management”
  - Used for lab monitor controller devices.

Without the correct role assigned, most devices will not function correctly.

3.3.1.2 Convert a Kiosk’s Role
The devices panel can be used to convert a check-in kiosk to a check-out kiosk or vice-versa. To
do this, first, navigate to the edit page of the device you wish to convert. Modify the role field
to one of the roles specified in Section 3.3.1.1 above. Next, visit https://sms.mfelabs.org a click
on the SimpleMDM application. If you do not see this button, you may not have access to
remotely control the kiosk devices. Access can be granted from the AWS SSO console. Once in
the SimpleMDM panel, click the device link in the sidebar, check the box next to the device you
wish to convert, and click restart in the action menu. Alternatively, a user could manually
restart the kiosk by opening the kiosk enclosure and holding the home and power buttons
simultaneously. These instructions only apply to the conversion between two kiosk roles.
Switching between a lab management device to a kiosk requires the installation of a different
application on the device.

3.3.2 Lab Users
The lab users panel provides a list of all users in the system. Do not delete any lab users unless
you have ensured there are no floating references to that user. Other entities such as shop
sessions reference a user ID. By deleting a user, historical usage information can be lost. Instead
of deleting users, delete their shop authorizations. The only valid case for deleting a user is to
remove duplicate entries caused by a user incorrectly entering their WPI ID on the basic user
training quizzes.
3.3.2.1  User Lost ID
When a user loses his WPI ID, WPI ID Services will create a new ID card for the user. The user’s WPI ID number will remain the same; however, the encoded RFID number will change. When a user with a new ID attempts to check-in, the kiosk will prompt him to enter his WPI ID number and then display an error stating a user with that ID number already exists. To clear this error, find the user in the Lab User panel through the search field. Press the edit button on the right side of the list. Scroll to the RFID number field and delete its content such that it is blank. Then press the “Update Lab User” button at the bottom of the form. Once this is completed, a user can successfully re-register by attempting to check-in. This will walk the user through the initial setup process.

3.3.2.2  Fix an Incorrectly Entered WPI ID Number
If a user incorrectly enters his WPI ID number in the “Basic Information” quiz in the basic user training, he will be unable to correct it himself. This is commonly discovered when a user tries to check-in for the first time, and the system displays an error stating they have not taken the basic user quiz. To resolve this problem, search for the user’s name in the search field of the Lab User panel. Ensure there is only one record of that user. If there are two, delete the record that does not have a shop authorization. Press the edit button next to the user and manually correct the user’s WPI ID number in the “Swipe number” field. The user should then be able to register successfully.

3.3.2.3  Change a User’s Role Display Name
User roles are displayed on the entryway signage when a user is the active lab monitor as well as on the badges that are printed when a user checks in. By default, the role displayed will be one of the following:

- Visitor
- Basic User
- Project Space User
- Lab Monitor
- Authorizing Lab Monitor

In cases where a custom role is desired such as “Washburn Staff,” a customized display name can be entered in the “Role display name override.” If this field is left blank, the default role will be used. Changing this value will not modify the role shown on the lab monitor controller device.

3.3.2.4  Change a User’s Email Address
An administrator can update the email address associated with a user from within the “Lab Users” tab of the administration panel. To do this, find the user to update in the list on the lab user list view and click the “edit” button next to his name. Next, change the email address in the “email” field of the lab user edit form. The email address of users with administration panel access must be changed in both the admin panel and the AWS SSO console. Should an
administrator only change their email address in one location, he will be unable to access the admin panel.

3.3.2.5  Badge Printing Behavior
By default, whenever a user checks-in on a kiosk equipped with a badge printer, a badge will be printed. This behavior can be changed to allow for cases such as the use of reusable badges for high-usage users such as staff members. To disable badge printing for a specific user, uncheck the “Should print badge on check-in” checkbox on the edit lab user form. This setting is universal and will make it such that a badge never prints for a user at any shop. At this time, disabling badge printing at only one shop is not implemented. This could be added in the future by moving this field to the Shop Authorization model.

3.3.2.6  Set a Custom Profile Photo
Whenever a user first checks-into a shop, he will be prompted to take a photo. If this user is a lab monitor, this picture will be used as his display picture on the lab signage. Users can retake their photos from the lab monitor controller by clicking the button in the upper right corner of their profile page and selecting the “Take New Photo” option. Alternatively, if an administrator wants to upload a higher quality profile picture, he can do so through the administration panel. To do this, the administrator should navigate to the user’s edit page and upload a file in the “profile picture” field. The server will process this picture and crop it to a square taken from the center of the image.

Should a picture be deemed inadequate, an administrator can force a user to retake their photo upon check-in by following the steps for lost IDs, outlined in Section 3.3.2.1 above, which will force the user to re-register and re-take his photo when he next checks-in.

3.3.2.7  Change User Permission
Each user has a permission level associated with their authorization to utilize a shop. To convert a basic user into a lab monitor or project space user, first, go to the user’s page from the “Lab Users” tab. On this page, there is a section labeled “Shop Authorizations.” This should show a list of shops the user has ever had an authorization to utilize. If the list is empty, it means the user has never been authorized in any shop. To create a new authorization, see Section 3.3.5. Once you have identified the user’s current authorization for the shop you wish to make them a lab monitor in, click the “edit” button next to the shop authorization. This will bring you to the shop authorization edit page. Here, you can change the user’s permission level by modifying the value in the permission field to one of the following options:

- basic_user
- project_space_user
- lab_monitor
- authorizing_lab_monitor
Should an administrator wish to make a user an authorizing lab monitor and for them have access to the admin panel, he will also need to create an account in the AWS SSO console with the same email address associated with the lab user.

3.3.3 Quizzes and Quiz Groups
Quizzes are one of the key features of the Washburn Lab Management System. The quiz feature enables users to complete online quizzes to gain access to shops or tools in the Washburn Shops. These quizzes and their various permissions are managed in the administration panel and in the Typeform interface.

Quiz objects are associated with a specific Typeform quiz and store information such as the maximum score.

Quiz Groups represent a group of quizzes where if all quizzes in the group are completed, a user is granted one or more shop privileges such as shop authorizations, tentative tool authorizations, or approved tool authorizations.

3.3.3.1 Create a New Quiz
Creating a new quiz is a multi-step process involving the use of the Washburn admin panel and the Typeform admin panel.

Create the Quiz in Typeform
The first step to deploying a new quiz is to create the quiz in Typeform. Typeform can be accessed at https://typeform.com, and the login credentials are available in the AWS Parameter Store. The quiz should make use of the calculator and conditional flow features to make it such that if a user doesn’t get all the questions correct, he is unable to move on to the next quiz. There should be two “Thank You” screens, one informing the user he didn’t pass with a retry button and one with a link to the next quiz in the quiz group. All typeform quizzes need to have a hidden field named “id_number” in order to track which user took the quiz. See the basic user quizzes in Typeform for an example of what a fully completed quiz should look like. Prior to adding the quiz to the Washburn Lab Management System, admins should verify the quiz functions properly as the backend will not add any functionality to the quiz.

Configure Webhooks
Whenever a user completes a quiz, Typeform will send an HTTP request called a webhook to the backend servers informing them that a user has completed a quiz. Typeform supports both regular and secured webhooks. A regular webhook provides no mechanism for verifying that a request from Typeform is legitimate. Typeform recently announced the support of a new webhook mechanism that utilizes a shared secret to enable the backend to verify the authenticity and integrity of a Typeform webhook request. We make use of this digitally signed mechanism to prevent the manipulation of quiz results. At this time, the only way to configure secure webhooks is through the Typeform API. To enable webhooks for a quiz, retrieve the Typeform secret from AWS Parameter Store and the Typeform API key from the Typeform...
Settings Page. Then execute the following command from a bash console with cURL (use Linux, Mac OS, or PuTTY):

curl --request PUT \
--url https://api.typeform.com/forms/{FORM ID HERE}/webhooks/phoenix \
--header 'authorization: Bearer {API KEY HERE}' \
--header 'content-type: application/json' \
--data '{
        "url": "https://sms.mfelabs.org/api/v1/webhooks/typeform/submitted",
        "enabled": true,
        "secret": "{SECRET HERE}"
    }
}

The form ID can be retrieved from the Typeform API or by copying it from the URL of the form editor page.

Add to Washburn Backend
The final step to adding a new quiz is to create a Quiz object in the administration panel. To do this, click on the “Quizzes” tab and then click the “New Quiz” button in the upper right corner of the screen. Fill in the form details according to Table 3.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>External Identifier</td>
<td>Enter the Typeform ID gathered from the form edit URL.</td>
</tr>
<tr>
<td>External Application</td>
<td>“typeform”</td>
</tr>
<tr>
<td>Display Name</td>
<td>The name to show in embedded quiz views.</td>
</tr>
<tr>
<td>Max Score</td>
<td>The maximum score a user can achieve on the quiz (in points). Any score below this will result in the system recording the attempt as a failure.</td>
</tr>
<tr>
<td>Quiz Groups</td>
<td>Leave empty.</td>
</tr>
<tr>
<td>Allow Anonymous</td>
<td>If checked, users will be able to take this quiz even if they don’t have an account. This is primarily used for quizzes such as the basic information quiz that create a new account for a user.</td>
</tr>
</tbody>
</table>

Table 3. The expected contents of a quiz’s fields.

3.3.3.2 Create a New Quiz Group
Once an admin has created a new quiz, he can create a quiz group which will set the permissions the completion of a quiz grants. This can be done from the “Quiz Groups” tab. Once on the quiz group list page, click the “New Quiz Group” button in the upper right corner of the screen. The fields in the quiz group form should be filled in according to Table 4. It is not necessary to specify both shops and tools in a quiz group, and only the resources that are granted by the quiz group should be entered.
When a user completes all the quizzes in this quiz group, a Shop Authorization object will be created for each shop in this field at the permission level entered in the “ permission level” field. A Tool Authorization object will be created for each tool listed in this field. If the Tool’s “auto_approve” field is true, the authorization will be marked as approved; otherwise, it will be marked as tentatively approved and appear on the lab monitor controller device in orange (See Section 3.3.7.1 for instructions on how to approve tentative tool authorizations).

Contains the list of quizzes that need to be passed to fulfill this quiz group. Quizzes can be found by typing a quiz’s display name or ID number.

The permission level that will be utilized for any shop authorizations created (See shops field).

The name that will be shown when displaying the quiz in the embedded quiz group view or other interfaces.

Table 4. The expected contents of a quiz group’s fields.

Once created, the Quiz Group ID number shown can be utilized in embedded quiz group views (See Section 4.3.2 of the Washburn Shops Management System MQP Paper).

Upon the creation or modification of a quiz group, users that have already completed the new requirements will not automatically be granted the associated permissions. To retroactively apply a quiz group, see Section 3.4.3.1.

3.3.3.3 Add a Quiz to a Quiz Group
To add an existing quiz to a quiz group, visit the quiz groups list page and find the quiz group you wish to add the quiz too. Next click, the “edit” button next to the quiz group. Type the name or ID number of the quiz being added in the “Quizzes” field. Adding a quiz will not revoke privileges already granted by the quiz group; however, it will affect all future completions.

3.3.4 Shops
The Washburn Lab Management System is designed to be a multi-tenant system meaning multiple facilities can use a single backend instance. To enable this functionality, Shop objects are used to store information about a specific facility using the system.

3.3.4.1 Create a New Shop
To create a new shop, navigate to the shops tab of the administration panel and click the “New Shop” button in the upper right corner. Fill in the form according to Table 5.
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Name</td>
<td>The name to when referring to the facility in user interfaces and on digital signage displays.</td>
</tr>
<tr>
<td>Responsible Lab User</td>
<td>Leave this field empty. This field represents the current active lab monitor for the shop.</td>
</tr>
<tr>
<td>Open Message</td>
<td>The message to show on digital signage displays when the facility is open. Line breaks can be represented by typing “\n.”</td>
</tr>
<tr>
<td>Closed Message</td>
<td>The message to show on digital signage displays when the facility is closed.</td>
</tr>
<tr>
<td>Lab Users</td>
<td>Leave this field empty. This field represents the users currently in a facility.</td>
</tr>
<tr>
<td>Tools</td>
<td>The tools that are present in a facility. These can be added upon creation or added later if Tool objects haven’t been created yet (See Section 3.3.6).</td>
</tr>
<tr>
<td>Cloudprint OAuth</td>
<td>This field contains the OAuth token used to authenticate with Google Cloud Print’s servers when printing badges. The system currently only uses the value from the Shop with the lowest ID number for all shops right now; however, this value can be copied from another shop record.</td>
</tr>
</tbody>
</table>

Table 5. The expected contents of a Shop object’s fields.

3.3.5 Shop Authorizations

Shop authorizations represent a specific user’s authorization to be in a specific shop. The authorization record stores information such as the user’s permission level and the expiration date of the authorization. Shop authorizations are usually automatically created by quiz groups; however, they can be manually created or modified by administrators. An easy way to access the authorizations for a specific user is to view the list of shop authorizations on a lab user’s show page (See Section 3.3.2.7 above).

3.3.5.1 Authorize a User to Utilize a Shop Without Completing Quizzes

Prior to manually authorizing a user, visit the user’s show page in the lab user tab to ensure that the user does not have any other preexisting and unexpired authorizations for the shop they are being authorized in. Conflicting authorizations can cause unexpected behaviors. Should a preexisting authorization be found, this authorization should be modified instead of creating a new authorization. Alternatively, the old authorization can simply be expired.

After verifying the user is eligible for the new authorization, visit the “Shop Authorizations” tab of the administration panel. Click the “New Shop Authorization” button in the upper right corner of the page. Fill in the form according to Table 6.
### Field Name | Value
--- | ---
**Lab User** | The lab user being authorized. The values in this list are sorted in ascending ID order so retrieving the lab user’s ID from the lab users tab can often save time in identifying users.
**Shop** | The shop to authorize the user for.
**Permission** | The level of access a user will have in the selected shop. Valid options are as follows:
- basic_user
- project_space_user
- lab_monitor
- authorizing_lab_monitor
**Expires At** | The date and time in which an authorization will no longer be valid.

Table 6. The expected contents of a Shop Authorization object’s fields.

#### 3.3.6 Tools
A tool object represents a tool or class of tools located in a specific shop. Tools can be added to quiz groups to automatically create authorizations upon the completion of a group of quizzes.

**3.3.6.1 Create a New Tool**
To create a new tool, navigate to the “Tools” tab of the administration panel. Next, click the “New Tool” button in the upper right corner of the screen. Fill in the form as described in Table 7. Tools can also be added to a shop by modifying the “tools” field of a Shop object.

| Field Name | Value |
--- | ---|
**Name** | The name that will be used to refer to the tool. |
**Shop** | The shop in which the tool is located. |
**Auto-Approve** | If true, Tool Authorization objects will be automatically approved when a quiz group with this tool is completed. If false, the completion of a quiz group will generate a tool authorization; however, it will be tentative. |

Table 7. The expected contents of a Tool object’s fields.

#### 3.3.7 Tool Authorizations
Tool Authorization objects represent a lab user’s authorization to use a specific tool. If a user has an authorization for a specific tool, it will show on any ID badges he prints and other information devices.

**3.3.7.1 Create New Tool Authorization**
First, navigate to the “Tool Authorizations” tab of the administration panel. As with Shop Authorization objects, it can be verified that a user does not have duplicate tool authorizations before adding a new authorization from the lab user show page. To create a new Tool
Authorization object, click the “New Tool Authorization” button in the upper right corner of the screen. Fill in the form according to Table 8

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool</td>
<td>The tool to authorize the user on.</td>
</tr>
<tr>
<td>Lab User</td>
<td>The lab user being authorized. The values in this list are sorted in ascending ID order so retrieving the lab user’s ID from the lab users tab can often save time in identifying users.</td>
</tr>
<tr>
<td>Approved</td>
<td>If true, a user will be able to use this tool immediately. If false, the authorization will be considered tentative and will require an administrator to approve the authorization.</td>
</tr>
</tbody>
</table>

Table 8. The expected contents of a Tool Authorization object’s fields.

3.3.7.2 Approve Tentative Tool Authorization

Tool authorizations have two states: approved and tentative. Approved tool authorizations allow a user to unconditionally use the tool specified in the authorization. Tentative authorizations are used to indicate that a user has completed a portion of the training for a tool but is still not allowed to use the tool. This is commonly used for the situation in which a user needs to take a quiz online and perform an in-person interview to gain access to a tool. Once the user completes the online training, he is given a tentative approval which an administrator converts to a full-approval after an interview has been conducted.

To approve a pending tool authorization, locate the tool authorization record on a lab user’s show page by selecting the appropriate authorization under the “Tool Authorizations” heading. Next, click the “Edit Tool Authorization” button in the upper right corner of the screen. Check the box labeled “approved” then press “Save Tool Authorization.” The user is now approved to utilize the tool specified in this authorization.

3.4 Interactive Console

While the administration is an essential tool for every-day administrative tasks, sometimes it will be necessary to perform more advanced operations in which the current interface will not be sufficient. For these operations, the system provides direct console access which enables users to execute scripts directly through an interactive console. While more complicated, the console provides an incredible amount of flexibility by exposing low-level components of the system. This allows users to perform tasks such as executing database queries and starting background jobs directly.

3.4.1 Console Access

To access the console, the following requirements must be satisfied:

- The user’s computer must have Ruby on Rails installed (https://rvm.io).
- The user must have access to a bash or an equivalent command prompt.
- The user must have the source code for the backend from AWS CodeCommit downloaded.
- The user must have sufficient AWS IAM permissions to access the production database keys stored in AWS Parameter Store.
- The user’s computer must have the AWS CLI configured and logged in (https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-install.html).
- The user’s IP address must be whitelisted on the RDS instance (this can be done by running “sh whitelist_ip.sh” from the source code directory).

Once all these requirements are fulfilled, the user can navigate to the directory where the backend code is located in his command prompt and enter the following command:

```
DISABLE_SPRING=true RAILS_ENV=production bundle exec rails c
```

![Image](image.png)

**Figure 14.** A screenshot after the successful creation of a production console.

If the user has successfully configured his machine, the console should look similar to that shown in Figure 14. At this point, the user can execute commands directly. To exit the console, type “exit” and press the return key.
3.4.2 Query Interface

The console is a full ruby interactive shell and can be used to execute any valid Ruby. The Washburn Lab Management System makes use of the ActiveRecord query interface to query data from the database. An ActiveRecord connection can also be used to execute SQL directly; however, this is discouraged. A full introduction to ActiveRecords can be found at the following link: https://guides.rubyonrails.org/active_record_basics.html.

Instead of providing a detailed introduction to ActiveRecord, this section will provide example queries that demonstrate how the query interface can be used. All the model definitions and the model methods can be found in the app/models/ directory of the source code. Additional documentation is also available in the Washburn Lab Management System MQP paper in Section 4.4.

**Get a Lab User**

lab_user = LabUser.find_by(display_name: “Sam Baumgarten”)  
lab_user = LabUser.find_by(email: “sbaumgarten@wpi.edu”)

**Get a Shop**

shop = Shop.first  
shop = Shop.find_by(id: 1)  
shop = Shop.find_by(display_name: “Washburn”)

**Count the Number of Lab Users**

LabUser.all.count # total number of lab users  
LabUser.where("email LIKE ?", "%@wpi.edu").count # lab users with WPI email address  
Shop.first.basic_users.count # the number of basic users authorized to access the first shop

**Aggregate Information**

emails = Shop.first.basic_users.map(&:email) # Gets the email address of each basic user  
Shop.first.lab_users.map(&:display_name) # Gets the names of all users currently in the shop
3.4.3 Tasks from Console
This section will provide example code for performing a few common tasks from the console.

3.4.3.1 Reload Shop Authorizations
If you want to retroactively re-calculate every lab users’ authorizations based on changes made to a quiz group, you can run the following command:

```ruby
quiz_group = QuizGroup.find(10) # finds quiz group with ID 10
LabUser.all.each do |lab_user|
  ReloadShopAuthorizations.call(lab_user, quiz_group)
end
```

This command may take a while to run as it iterates over every lab user in the system.

3.4.3.2 Expire All Current Shop Authorizations
This command can be used over the summer when all shop authorizations from the prior year need to be expired.

```ruby
shop = Shop.find(1)
shop.authorizations.each do |authorization|
  authorization.expires_at = DateTime.now
end
```

After running this command, all users will need to be re-authorized to continue accessing the shop. If only the code snippet above is executed, users will regain access after taking only one quiz as their old quiz results will still exist. This means they will still satisfy the quiz group providing shop authorizations requirements and automatically be authorized again when their permissions are reloaded. To solve this issue, all old quiz results will need to be deleted. This can be achieved by executing the following command:

```ruby
QuizResult.destroy_all
```

**WARNING**

This should not be used when multiple shops are using the same backend unless all shops are being reset. Running this command will delete quiz results for **ALL** shops.

3.4.3.3 Send Message to Current Lab Monitor
The following command can be used to send a push notification to the current lab monitor:

```ruby
SendAlertToLabManagementJob.perform_later(Shop.first.id, 
"[Your Name]: Type your message here."
)
```
4 Conclusion and Future Work

This IQP developed tools that complement the Washburn Shops Lab Management System, enabling advanced data comprehension, data-based decision making, and simplify the process of interacting with the backend system. Our analysis of various business intelligence (BI) tools identified a suitable BI tool for the current needs of the facility staff while also providing a base of knowledge that will be relevant should future upgrades be necessary due to changing requirements. We selected Google Data Studio as the current data comprehension tool while also providing extensive resources for a future upgrade to Microsoft Power BI should it become necessary. In addition to the selection process, we also developed example visualizations that provide a starting point for future reports while also be useful as-is. The visualizations developed in this IQP have been deployed in the Washburn Shops and have already been utilized by the staff to better understand how the facilities are utilized and make meaningful policy changes. An example of a policy change that was made based on the reports was a change to some of the basic user training quizzes where the reports identified specific quizzes that were taking significantly more attempts than other quizzes to complete. When this was brought to light by the reports, the training material was changed to provide more information regarding the area many people were having problems with.

In addition to the data comprehension tools analyzed and developed as part of this IQP, we built a tool for the manipulation of data recorded by the Washburn Lab Management System. Prior to the data manipulation/administration panel being built, all errors were corrected manually with no mechanism for shop staff to make changes directly. This meant users and shop staff had to wait for changes to be made which could cause inconvenient delays if information was incorrectly entered or when situations such as lost WPI IDs occurred. The administration panel solves this problem by enabling shop staff to make common fixes to the data stored by the system directly without a significant amount of technical knowledge regarding the inner-workings of the system. This paper provides instructions on how to complete many everyday tasks that may need to be done from the administration panel and describes how to gain access to a production console enabling advanced data manipulation tasks.

This project establishes a significant foundation for the Washburn Lab Management System in the areas of data comprehension and data manipulation, but it also leaves a significant amount of room for growth. We focused on single data source visualizations, only pulling data from the backend system; however, there is a considerable amount of room for discovery in the development of new visualizations that combine data from different data sources. Future analysis could help the shop staff not just understand how the manufacturing labs are utilized, but also assist in identifying factors that are correlated with long-term use of the facility. This analysis could help refine the teaching material for courses such as ME 1800. Other opportunities for future work include the automation of maintenance tasks such as automated certificate renewal. Annual maintenance tasks for this system tend to be repetitive, and a system to automate them could have applicability outside the scope of this project.