Journal of Information Systems Education

Volume 33
Issue 3
Summer 2022

Teaching Case
Widgets-R-Us: Using IoT to Monitor Part Levels

Paul Dunn and Robert E. Miller

Recommended Citation: Dunn, P., & Miller, R. E. (2022). Teaching Case: Widgets-R-Us: Using IoT to Monitor Part Levels. *Journal of Information Systems Education*, 33(3), 229-231.

Article Link: https://jise.org/Volume33/n3/JISE2022v33n3pp229-231.html

Initial Submission:	20 May 2021
Minor Revision:	6 October 2021
Accepted:	20 October 2021
Published:	15 September 2022

ISSN: 2574-3872 (Online) 1055-3096 (Print)

Journal of Information Systems Education, 33(3), 229-231, Summer 2022

Teaching Case
Widgets-R-Us: Using IoT to Monitor Part Levels

Paul Dunn
Robert E. Miller
Business Information Systems Department
Central Michigan University
Mt. Pleasant, MI 48859, USA
dunn1pt@cmich.edu, mille5re@cmich.edu

ABSTRACT

This teaching case describes the functionality of a system employing IoT sensors to monitor part levels for a fictitious auto parts manufacturer. Data from the sensors are used to populate a centralized database and generate a dashboard for management. The system also generates tiered alerts to notify part runners and managers of pending part shortages. The case is based on an actual systems development project that students completed for a real auto parts manufacturer. Material in the case can be used for assignments in systems analysis and design, database management, and data analytics courses. Teaching notes, including an entity-relationship diagram (ERD), data dictionary, data flow diagrams, prototype dashboard, and discussion questions are available through the JISE website.

Keywords: Database design & development, Entity-relationship modeling/diagram, Systems analysis & design, Data flow diagram, Data analytics, Teaching case

1. INTRODUCTION

Widgets-R-Us is a small, midwestern, auto parts supplier. Founded in 1967, Widgets-R-Us specializes in Original Equipment Manufacturer (OEM) products for auto manufacturers such as Ford, GM, FCA, and Honda. Widgets-R-Us produces three products (A, B, and C) in its factory in Mount Pleasant, Michigan. Each product is manufactured on its own assembly line on the factory floor. The assembly of each product is a multi-step process involving multiple workstations and numerous component parts. As an established company in the auto parts market, Widgets-R-Us has long had a reputation for producing quality products in a timely manner. Unfortunately, this reputation has recently come into question as the company has missed some deliveries due to a slow-down in production. This slow-down is directly related to the company's addition of Product C – its newest product line. Since Product C was introduced, overall production levels have dipped causing the company to miss delivery deadlines. To address this issue, Widgets-R-Us hired a new production manager who has just completed a review of the factory's manufacturing process. Based on this review, the manager believes she has determined the source of the problem, namely, the current process used to resupply parts to the workstations. The manager has decided to rework this process by implementing Internet of Things (IoT) sensors and a real-time dashboard to collect and display data about part levels in the three production lines. These changes are expected to cut the number of part stock-outs, resulting in increased production at the factory and a return to timely deliveries.

2. BACKGROUND

For this case, the students will assume the role of system analysts responsible for designing and implementing a system to capture, display, and report parts data from the factory's production lines. Data will be captured in real-time using IoT sensors. The captured data will be displayed using a dashboard simulating the manufacturing floor. The dashboard will also allow the production manager to generate reports from the system database. Finally, the system will notify employees and management of impending stock-outs so they can resupply parts proactively.

3. CASE TEXT

3.1 Overview of the Current Workstation Replenishment Process

The Widgets-R-Us factory operates 24/7 with three shifts: day, mid, and night. The factory has three assembly lines, one for each product. Each of the lines has multiple workstations. At each of these workstations there can be any number of racks. These racks hold the parts that the employee at that workstation uses for their portion of the assembly process. To simplify its inventory, Widgets-R-Us puts all parts used at a workstation in the racks, whether parts by the piece or in bulk. Each rack holds up to three trays of a specific part. The trays are organized in the rack so one tray is available for the worker to pull parts from while the other trays are held in reserve. When a worker empties the tray he has been pulling parts from, the empty tray is removed and the reserve trays roll down the rack, making a new tray available to pull parts from.

As an example, for Product A on Line 1 there are three workstations. The first workstation has two racks. The first rack contains trays for Widget-A housings, 10 housings per tray. The second rack contains trays for Widget-A motors, 8 motors per tray. The worker at this station would assemble the housing and the motor and send that sub-assembly to the next workstation. The second workstation on Line 1 has two racks. The first rack contains trays for Widget-A covers, 20 covers per tray. The second rack contains trays with rubber seals, 100 seals per tray. When the worker at this station gets the sub-assembly from the first workstation, their job is to put a rubber seal in a cover and affix the cover to the sub-assembly. After completion of this operation, the sub-assembly is sent to the last workstation. The last workstation on Line 1 contains two racks. The first rack contains trays with M10- 1.25x50 flanged hex head bolts, 100 bolts per tray. The worker uses 5 bolts to affix the cover to the housing. The completed product is then placed in a tray in the second rack. Currently, Widgets-R-Us has "part runners" whose job is to walk the manufacturing floor, checking to see which workstations need to be resupplied with parts. Given the number of lines, workstations, and racks, this can be challenging. In order to help the part runners, the racks have been painted with a different color for each position of a tray. Since it is possible to have three trays on a rack, the tray positions are painted red, yellow, and green (see Figure 1).

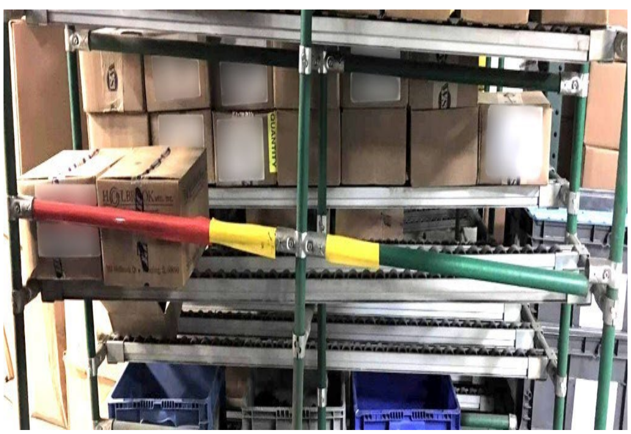


Figure 1. Rack with Tray Positions Marked

The first position is painted red. This is for the tray closest to the worker – the tray parts are being pulled from. The second position is painted yellow. This is for the first reserve tray. The third position is painted green. This is for the second reserve tray. If the part runners check a rack and see the last tray in the green position, that means the rack is full (one tray being used with two trays in reserve). If the rack's last tray is in the yellow position that means the rack is down to one reserve tray. If the rack's last tray is in the red position that means there are no reserve trays left. Using this method, part runners know, at-a-glance, if they need to get parts.

Although the system has largely worked in the past, there have been an increasing number of issues since the introduction of Product C. Given that the production floor is full of equipment and people (more so with the new product line), it is not always easy for part runners to see each work station's racks clearly. Occasionally, a rack may get overlooked, especially if the part runners are busy in another part of the facility. This results in workstations that run out of parts, bringing the assembly process to a halt. This has directly led to the company missing delivery deadlines and reduced revenue. Even when part runners see that a rack is running low, they still have to physically check the rack to verify which specific part needs to be refilled – taking up more valuable time. For these reasons the new production manager has decided to rework the process by implementing a system that includes IoT sensors and a real-time dashboard with alerts and reports. The following sections detail how this new system will work.

3.1 IoT Sensors

One of the engineers at the factory put together a prototype that uses light sensors to detect if there are trays in place. This prototype uses Light-Dependent Resistors (LDRs) to detect if a tray is present over them (Figure 2).

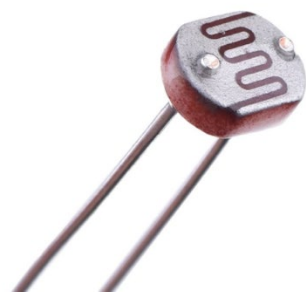


Figure 2. Light Dependent Resistor

As it is currently built, the prototype turns on lights that correspond to the colors on the racks. Green is used for full, yellow for one reserve left, and red for no reserve left. When all trays are in place, the green light is turned on. Since the LDRs are placed in series on the rack, as a tray is removed, the rest of the trays roll forward and the first LDR is uncovered. The circuit is designed to then turn the green light off and the yellow light on. As more trays are removed, and another LDR uncovered, the yellow light turns off and the red light turns on. This system, when installed on a rack, will automatically send data to the system database. This data will be used to update the dashboard and generate necessary alerts.

3.2 Dashboard

The system will provide a dashboard that the production manager can use to quickly check the parts levels at each workstation in the factory. The dashboard will display this information using a schematic diagram of the manufacturing floor. Each product line will be displayed with its workstations in the proper production sequence. Each workstation will display its racks. The racks will include colored lights (red, yellow, and green) to indicate the trays present. Each rack will also display the part it holds.

3.3 Alerts

The system will generate tiered alerts when part levels drop below acceptable levels. First-tier alerts will be sent when a rack has only one tray in reserve (condition yellow). These alerts will be sent via text message to the part runner currently on duty. They will identify the workstation and part that needs to be replenished, along with the rack into which the part tray should be placed. Second-tier alerts will be sent when a rack has no trays in reserve (condition red). These alerts will be sent to the part runner via text message and the production manager via email. Second-tier alerts will also be displayed on the production manager's dashboard. Finally, third-tier alerts will be sent when a rack is empty. These alerts will be sent to all available managers via email, in addition to text messages sent to the part runner and the production manager. Third-tier alerts will also be displayed on the production manager's dashboard. Third-tier alerts are obviously very serious since they mean that work on a product line has, effectively, come to a halt.

3.4 Reports

The system will provide a set of both canned and ad-hoc reports for use by Widget-R-Us' management. The students are tasked with defining and developing the reports based on the business needs described in the case and the specific data captured through their system design.

4. ASSIGNMENT

The Widgets-R-Us case can be used to develop a number of possible assignments. These assignments can be completed by individuals or groups. They can be used to create multiple homework assignments or one comprehensive system design project. The authors provide examples of possible assignments in Table 1.

Class Activity	Assignment	Deliverables
System requirements lecture, Review Widgets-R-Us case	Document system requirements	System requirements document
ERD lecture, Review Widgets-R-Us case especially factory entities (line, workstation, rack, etc.)	Develop data model	ERD
DFD lecture, Review Widgets-R-Us case especially input from IoT sensors and alerts to management	Model system processes	DFD
User interface design lecture, Review Widgets-R-Us case especially dashboard layout	Develop desktop prototype	Desktop application forms
IoT lecture, Review Widgets-R-Us case especially data collection	Document rules for when sensors should report	Program code to insert sensor data into database and generate tiered alerts
Data analysis lecture, Review Widgets-R-Us case especially performance metrics (number of stock-outs, time to replenish, etc.)	Analyze sensor data to produce performance metrics	Performance report with suggestions for improving the production process

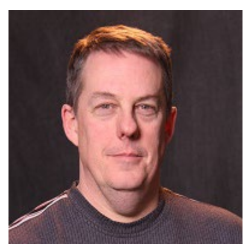
Table 1. Possible Assignments

5. CONCLUSION

The case presented herein provides numerous opportunities to engage students in a real-world system design problem. Students working as individuals, or in groups, can tackle assignments ranging from documenting requirements to designing databases and dashboards. Students can perform data analysis using IoT sensor data and even complete a hardware-based assignment to collect the data in real-time. The assignments can be completed as stand-alone exercises or combined to create a complete system design project.

The case is based on a systems development project that students completed for a real auto parts manufacturer. Widgets-R-Us may be fictitious but the scenario is not. IoT has become a significant part of the manufacturing industry and its use is only expected to grow. Students will benefit from their exposure to a real-world case and the challenges it presents.

AUTHOR BIOGRAPHIES



Paul Dunn is a lecturer in information systems at Central Michigan University. He received his Master's in Information Resource Management from Central Michigan University. He has over 20 years of experience in client/server programming, database design and administration, and system administration. His research interests are big data, NoSQL, and data analytics.

Robert Miller is a professor of information systems at Central Michigan University. He received his Ph.D. in Information Systems from the University of Arkansas. His current research interests include information systems service quality, social media, and faux pas posting. His publications have appeared in several journals including *Behaviour and Information Technology*, *Information Systems Management*, *Journal of Information Systems Education*, and *MISQ Executive*.

Information Systems & Computing Academic Professionals

Education Special Interest Group

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the ***Journal of Information Systems Education*** have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 2574-3872