Inventory System for Minor and Major Equipment

August Davis
Student
Department of Computer science
August.Davis@und.edu

David Erickson
Student
Department of Computer science
david.p.erickson@ndus.edu

Christian Hansen
Student
Department of Computer science
christian.hansen@ndus.edu

Abstract: *Currently the University of North Dakota Computer Science department (UND CSCI) tracks its equipment via an Excel spread sheet. Under the current system each piece of equipment is given a barcode representing a unique identification number which then then must be manually added to the inventory spread sheet. This is a time-consuming task combined with the access issue presented with storing this information in a single spread sheet file makes the current system inefficient and to an extent, risky to use. This project is aimed at replacing the current system with one that will allow multiple users appropriate access to the inventory data, as well as speed up the entry process by scanning each items barcode with a phone camera. To accomplish this, a web application was built to provide an interface for the user to access the different functions. From this application, they will be able to create, delete, view, update, and locate inventory given the correct permissions.*

INTRODUCTION: The departments at the University of North Dakota (UND) have two broad categories of inventory that must be kept track of: Major inventory, which consists of inventory costing over $5,000, and Minor inventory, consisting of inventory under $5,000. The Computer Science department currently handles tracking its inventory by giving each item a unique identification number and manually adding it with all the relevant information about the item to a spreadsheet in an Excel spreadsheet. Some of the other information currently includes the room number, serial number, description, quantity of the same item, date purchased, original costs, and replacement costs.

The lack of user access control and inventory tracking accuracy creates a security and logistics issue. With the current system, there is one central copy that can be copied to others, but only the original will be used to track any changes. If the spreadsheet should become corrupted, then the department could lose inventory records if they are not backed up.

The goal of this project is to create a similar centralized database where multiple users can securely access inventory information and update the database in a controlled method. The user will also be able to utilize the barcodes that already exist on current inventory to locate and update inventory in the database. This will improve the speed and accuracy of inventory management.

RELATED WORKS: *Microsoft Visual Studio* offers a well-rounded platform for us to develop our application. With their use of .NET, we will be able to create an application that can be accessed across Android, iOS, and windows. Visual Studios also offers the use of Azure App Service to connect the Universities own database.

Microsoft SQL Server will enable us to store our inventory data on-site at UND. It is a cross-platform database technology which stores and retrieves data as needed by the user. Its use is heavily integrated into Microsoft's other technologies, such as Azure and Visual Studio, allowing for easy usage. It can also pass reports to mobile devices, which will give us a way to display data that can easily be interpreted by the user.

Mobile User Experience Guidelines and Recommendations Digital Gov published this article which outlines some of the guidelines that have been deemed the most relevant from community events from 2013-2015. They have distilled that 1) make sure your content is structured and chunked appropriately for multiple devices. This is useful to make sure that one key aspect of our design will work across multiple devices without building a dependency on it just to find out it won't work correctly. 2) Follow industry user interface guidelines and government regulations in the development of your mobile product. For our project we will be using UND's Identity Guidelines. 3) Leverage the device s features for usability and accessibility. We will be making use of the camera to scan barcodes and the wireless connection to be able to connect to the database anywhere with the necessary login information. 4) Test at multiple points in the design and development process. We intend to continuously test and modify our code to ensure that it is stable and brings an easy experience to the user. 5) Collect and use data to determine what content your users want and where. This will be used in the later stages of our development when we present our prototype to test subjects to see what they do and do not like about it and where it could use improvement. 6) Develop security and privacy guidelines regarding what the app does and how it protects user data. Security will one key aspect since there will be remote access to UND's database.

University Of North Dakota (UND) The University of North Dakota is required to follow "certain identity standards" while still being allowed to be flexible with the web content. UND defines a set of style guides for web content that will be brought across into the design of our application such as the color scheme, consistency of the navigation bar, and formatting of lists. These are just a few of the things that they outline.

APPROACH: The team started working on this project in Visual Studio because of its .NET services to build a server-based web application. Visual Studio also allows testing with different web servers and provides easy integration of different packages such as a Scanner API. It also provides in depth debugging and controlled testing.

The previous year's team had built a series of applications which had worked, but due to the lack of documentation and segmented source code, it was not possible to continue work on their project. So, this year's team decided to rebuild the system from scratch and document the source code to provide an easy platform for future developers to maintain and update the system.

The first attempt at a solution was based around the idea of building a separate application for Android, iOS, and Web-based platforms. This introduced a development challenge in finding and building something that would work well between all platforms. The team eventually decided to pivot directions and focus on one web-based application that would work across all platforms and feed off the same code base. This was decided when it was realized that how we wanted to handle the barcode scanning would be very difficult to maintain across the different platforms. The team is also taking extra steps to make sure that the future developers of the system will be able to work on the project with minimal friction.

IMPLEMENTATION: The inventory system project was broken down into three main parts: user interface design, backend logic development, and database administration. User interface design began with researching the University of North Dakota's guidelines and standards for software and interface development. Specific colors and icon styles we mandated by the university to maintain consistency across all of the websites used by UND. It was important for the team to carefully follow these guidelines, as to not lead to more work in the future and to ensure immediate acceptance by the department and school. Ensuring the design is clear and appealing to look at is also very important. Users

must be able to quickly perform their tasks without delay, or acceptance and use of the application will remain very low.

When this project was going to be a stand-alone Android app, the plan was to integrate the scanner function into the app. When the focus pivoted to building a website, this became impractical due to how the ASP.NET framework processes information on the side of the server and being priced out of commercial solutions. The solution to this problem was to rely on external hardware and apps to handle the scanning. For PC, this means a handheld USB barcode scanner to scan the code in to the appropriate text box. For Android, the scan button calls an external app on the phone to achieve the same affect; an iOS version of this should be possible but is not currently one of the goals of the project.

The website portion of the project was created using the ASP.NET web framework. This framework combines the convenient features of HTML5, CSS, and JavaScript, allowing for a robust website that looks aesthetically pleasing to the user. CSS capabilities will help the team to follow the guidelines for color and design set in the UND design standards.

Microsoft SQL Server is being used for the database portion of the project. The security features present in this software will prevent use by any unintended users. The team will be using stored procedures, which are subroutines and queries that are stored in the database that are used to handle all of the database transactions. Data is passed to these stored procedures, and this data is then inserted into the database, updated with new data for a given record, deleted from the database table, or selected for viewing. User roles are implemented to give access to specific stored procedures, so only certified users will be able to perform transactions on the database.

Adding barcode scanner functionality is crucial for use by UND faculty. Having a user type in a complex code rather than take a picture of the barcode adds a layer of complexity and inconvenience which makes the application difficult and cumbersome to use. A utility called Spire.Barcode will be used for this purpose. This free API allows for users to take a picture of the barcode with a cell phone camera or webcam to scan in the code on the tag, which will increase the speed of both looking up equipment and adding new equipment. Use of a barcode scanner is the basis of this system, and without it the project will likely never be used by faculty.

In order to get the app running, a future team will need to follow the instructions created by the current team. Important utilities needed to be installed include the following: Microsoft SQL Server, Microsoft SQL Server Management Studio, Microsoft Visual Studio, Spire.Barcode, and the source code of the application itself. Instructions for setting up each of these tools will be included in the folder with the source code.

RESULTS: The pages currently work on PC systems, but remain to be tested on mobile devices. Mobile devices will be able to access the site from anywhere, but testing needs to be done to see how usable it is. On PCs, the website is fully functional and usable. The formatting and setup of the website works as intended.

The database tables are set up to contain all of the information needed to track and look up pieces of inventory. Use of stored procedures allows for users to quickly access, change, add, and remove inventory data. Future teams will be able to expand the database to be able to store more information to open up the system for use by other departments or the entire University. Different user roles can be added for each department to ensure faculty can only access inventory from their own department.

Barcodes will be able to be entered in multiple ways. The first way is to take a picture using a phone or digital camera, then transfer the image to a folder on the PC and scan the stored image. This method of scanning is not practical and slows down the operation of the website. Another way to scan barcodes on the PC would be to use a dedicated barcode scanner to input the barcode text into the website form. This would be the fastest way for faculty to upload data to the inventory system. On mobile devices, the easiest way to add barcode data would be to call another application to scan the code, and then pass the data back to the website.

Current documentation will be expanded to ensure future teams will be able to continue working on the system. Setup guides will be included, alongside descriptions of how each part of the project works. This will be satisfactory to show future developers how to start the project, and how to best expand it.

FUTURE WORK: There will be some work to do on the system after this year. Some things that still need to be addressed include where to host the database and how deploy it. The team was in contact with professor Nordlie to see about hosting the app on UND's servers but were not allowed. It was discussed with Professor Grant that they might try to host it on a machine in the CSCI department. This may be best short term, but it would be preferred to move to a larger server as a scale up in inventory occurred. Some extra features that could be added in the future is tracking the status of the inventory and adding an option to turn in a maintenance request if it breaks.

One feature that can be streamlined is the barcode scanning process on mobile. Calling a separate app to scan codes takes extra time and can slow the process of entering codes. Another feature a future team can add is iOS compatibility and to ensure both iOS and Android maintain full functionality. This will allow for anybody to use the system no matter what platform they are on.

CONCLUSION: The team has seen lots of both success and failure during the process of working on the project. At the start of the semester, the previous team's project was inaccessible due to lack of documentation and missing admin credentials. This resulted in the current team moving to start a new project from the ground up, creating a native Android app. After work was performed to get this app started, there was trouble getting the project running to make additional changes, so again the team switched the scope of the project to be a website. The present website system has been the easiest to use and develop, and appears to be the most accessible across the different platforms.

Future teams can build upon this project by solving the logistics issues associated with deploying the system. The database and web server are the main focus of this issue, as storing the system on a centralized web server will be crucial in the use of the system.