**Inventory Management System**

**Conor McGrath**

**Aaron Healy**

B.Sc.(Hons) in Software Development

November 7, 2018

**Final Year Project**

Advised By: Daniel Cregg

Department of Computer Science and Applied Physics

Galway-Mayo Institute of Technology (GMIT)



**Contents**

[Introduction 4](#_Toc6318713)

[Context 5](#_Toc6318714)

[2.1 Objectives 5](#_Toc6318715)

[Methodology 7](#_Toc6318716)

[3.2 Version Control 8](#_Toc6318717)

[3.3 Technology Choice 8](#_Toc6318718)

[3.4 Testing 9](#_Toc6318721)

[Technology Review 10](#_Toc6318722)

[4.1 Database 10](#_Toc6318723)

[4.1.1 MySQL 10](#_Toc6318724)

[4.2 Backend 10](#_Toc6318725)

[4.2.1 Java 10](#_Toc6318726)

[4.2.2 Spring 11](#_Toc6318727)

[4.3 Frontend 11](#_Toc6318728)

[4.3.1 Angular 11](#_Toc6318729)

[4.3.2 Typescript 12](#_Toc6318730)

[4.3.3 HTML 13](#_Toc6318731)

[4.3.4 CSS 13](#_Toc6318732)

[System Design 15](#_Toc6318733)

[System Evaluation 15](#_Toc6318734)

[Conclusion 15](#_Toc6318735)

**About this project**

**Abstract** This paper aims to provide a technical understanding about computerized inventory management systems; the various implementations of, and the technologies used within these systems. Modern businesses can utilize the latest technologies for their warehouses to help:

* analyze trends or patterns through inventory flow
* reduce instances where stock is too low or high, and reduce instances where there is no stock
* increase in accuracy and fulfillment of orders with relation to picking, packing, and shipping a customer’s order
* provide a deeper understanding of customer demand for your products

The significance of this study will redound to the benefit of consumers that utilize such systems, especially since businesses are faced with the problem of getting products and services to their costumers quickly and responsively. The demand for better customer experiences, justifies the purpose to evaluate the current technologies already available. Thus, businesses that integrate these technologies and solutions can create stronger customer experiences, through the tight integration of their web platforms and enterprise resource planning systems. For the researcher(s), this study will help advance our knowledge and evaluate the approach with which to take when determining which technologies are best integrated with one another when developing inventory management systems. Thus, a new implementation or integration for these systems may be established.

**Authors**

Conor McGrath - B.Sc. (Hons) Computing in Software Development

Aaron Healy - B.Sc. (Hons) Computing in Software Development

**Chapter 1**

# Introduction

This chapter will outline the objectives of the project along with the scope which we plan to complete those objectives in. An analysis of each of the chapters found in this dissertation along with a summary Github repository containing the project can be found below. This application will aim to satisfy the standards for a Software Development Level 8 project by surpassing the expectation of Inventory Management Systems currently offered online. Inventory Management Systems are a key part to any business. They allow the business to manage and control their customers, wholesalers, orders and stock. They are essential in giving up-to-date data on movement of stock throughout the company.

The members of this group have taken it upon themselves to research and analyze existing Inventory Management Systems available, and with the knowledge gained create an application that will allow users to manage all their customers, wholesalers, orders and stock in an easy and conventional way. The frontend of the application be presented using Angular 6 which will be running off a Java Spring Boot backend. Proper authentication and login will be applied to the app to allowing users to easily assess their current stock, current and past orders, wholesalers and their current customer base. All the data will be stored in a MySql database hosted by Amazon Web Service.

**Chapter 2**

# Context

The general context of this project is an easy to use system that provides users online sales businesses a platform to manage their inventory. It will help them keep track of all stock and alert them when they are running low on certain products. The users will be able to track their customers and view who purchases what number of products and how regularly. This will allow a business to see what sort of customer their main target for future sales is. The products section of the application will allow users to view and manage the current stock they have. This will show them what products are selling good and the profit they are making from each product. This sort of information is vital for a business to know so that they can see the good and bad sellers and order more inventory accordingly. In terms of the order side of the application, the user will be able to input orders so that they can be tracked. When these orders are completed the products and customer information will all be updated accordingly. Each user will have their own home page which will contain a variety of tables and graphs that represent their current monthly information such as “Total Sales”, “Top Selling Products” and “Top Spending Customers”.

## 2.1 Objectives

The main objective of our application is to help businesses to manage their inventory easier. We also wanted to make it easier for businesses to be able to view all the statistics for their sales information easier in one place. The following is a list of the main pages in our application along with the objectives for each page.

1. **Login/Register:** The first page the user will see is a login and/or register page. If the user is new, they will be able to create an account using the register function. If the user has already created an account, they will be able to login using their credentials. Once logged in, the user has access to all the features of the application. All users of the application must be registered so that they can have their inventory data linked to them and only viewable by them.

2. **Home:** The objective of the homepage is that it is a base of navigation for the application while also providing the user with quick available statistics about their inventory. The homepage will display a variety of graphs and tables that will show information regarding the current month of sales information. The homepage will also provide links for the user to follow to the other pages of the application.

3. **Customers:** The customers page will provide a base for the user to view all information about their current customers. This will include information such as the products they have purchased and in what amount. Users will also be able to create, edit and delete customers on this page.

4. **Orders:** The orders page will be the location where the user will be able to manage their current orders, create new orders and look back on previously completed orders. The user will also be able to view orders from specific customers which will allow them to see what each customer is ordering and in what amount.

5. **Products:** The products page will provide a base for the user to view all information about the current products they have to offer. This will include information about the products such as cost price, sale price and the quantity in stock. This is vital information as it will allow the user to see what sort of margin they are making on products and when a re-order of a certain product is needed.

**Chapter 3**

# Methodology

This is where will be discussing the methodologies we are using in our project. We researched the different methodologies we could use in our project. After seeing the positives and negatives of each methodology, we decided to use Agile as our main methodology. We felt this was a methodology that would suit our development and it is also a methodology that is used widely in organizations around the world

**3.1 Agile Development**

Our project contained three main stages, research, design and implementation. For each of these stages we applied an Agile like approach to complete them. Agile was suited to this project as it allows for flexibility and for us to deliver software incrementally. This is what made it stand out to us as a methodology that would work well for us. We used a Scrum like approach for our research, design and development process. Scrum is an agile framework for teams who members break their work into actions that can be completed within timed iterations, called sprints, usually around two weeks long but no longer than one month, they then track progress and re-plan in short meetings.

Our sprints involved certain parts of the development being completed. Our first sprint in the development phase was to create the database that would be used for storing our data. After this our sprints followed a pattern of developing a backend and frontend component for each part of the project.

This would mean one sprint would involve developing the users, another, the customers, the next, the products and so on.

We held weekly meetings with our project supervisor, this allowed us to have meetings before, after and during our sprints. This constant contact with our supervisor was key to making sure we were completing everything on time and to make sure the project was running smoothly. After each of these meetings we were able to note what was needed to be completed before the next one. This gave us a goal to work towards each week in our sprints.

Any issues we noted at a meeting were logged into our Github repository under the issues tab. This allowed us to track problems we had throughout the problem.

!!NEED TO INSERT IMAGE OF ISSUE LIST!!

## 3.2 Version Control

Version control is key in any project development cycle. It is important as it always allows members of the team to work on the project at the same time and keep a working version of the project in a safe place. We decided to use Git and Github for our version control.

**3.2.1 Git**

Git is a distributed version-control system for tracking changes in source code during software development. It is designed for coordinating work among programmers and can be used to track changes in any set of files. Git’s best features include its speed, data integrity and support for distributed, non-linear workflows.

As with most other distributed version-control systems, and unlike most client–server systems, every Git directory on every computer is a full-fledged repository with complete history and full version-tracking abilities, independent of network access or a central server. Some of the main features of Git are:

**Strong support for non-linear development**

Git supports rapid branching and merging. It includes specific tools for visualizing and navigating a non-linear development history. Git works on the basis that a change will be merged more often than it is written. Branches in Git are usually very lightweight: a branch is only a reference to one commit.

**Distributed development**

Git gives each developer a local copy of the full development history, and changes are copied from one such repository to another. These changes are imported as added development branches and can be merged in the same way as a locally developed branch.

**Efficient handling of large projects**

Git is very useful for use with large projects. It is very fast and scalable, and performance tests completed have showed that it was an order of magnitude faster than other version-control systems.

**Pluggable merge strategies**

As part of its toolkit design, Git has a well-defined model of an incomplete merge, and it has multiple algorithms for completing it, culminating in telling the user that it is unable to complete the merge automatically and that manual editing is needed.

**Garbage accumulates until collected**

Aborting operations can leave useless dangling objects in the database. Git will automatically perform garbage collection when enough loose objects have been created in the repository.

**3.2.2 Github**

GitHub is a web-based hosting service for version control using Git. It offers all the distributed version control and source code management functionality of Git as well as adding its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project. These extra items were very useful for us developing this project.

Github is one of the most popular version control management services in the world. As of June 2018, GitHub had over 28 million users and 57 million repositories which included 28 million public repositories, making it the largest host of source code in the world.

## 3.3 Technology Choice

After researching Inventory systems and what type of application we would be designing, we made our decisions about the languages and software we would be using for our project.

We decided on a MySQL database hosted on AWS for the database side of our project, Java Spring Boot for our backend and Angular 6 for our frontend. These technology choices were made based on our experience using each and the advantages they gave us over the other options. Each of these technologies will be explained in the Technology Review chapter.

**Backend Frontend**



## 

## 3.4 Testing

Testing is a vital part of every project. It is essential that tests are carried out on a project to make sure every aspect of it is working in the correct way. There are many types of testing which we researched for using in this project. In the end we decided to test our project using the below techniques.

!!NEED TO FINISH TESTING!!

**Chapter 4**

# Technology Review

In this section, we will review all the different technologies used in our project. We will discuss what each technology is, and what it does in our project. We will use images and code snippets to show how the technology is implemented into our project. We will split the review into three smaller sections which will be the Database, Backend and Frontend sections. We will talk about the technologies used in each section and why we chose to use each for that section.

## 4.1 Database

### 4.1.1 MySQL

MySQL is a free-to-use, open-source database that facilitates effective management of databases by connecting them to the software. It is a stable, reliable and powerful solution. The choice of using MySQL as our database for our project was easy to make. Both members of the group have used MySQL for several modules over the course of our degree. We also used MySQL in our 3rd year project. This experience gave us the confidence to use it in our project. We also felt MySQL would suit our project for several reasons, which we mention below.

MySQL is globally renowned for being the most secure and reliable database management system and is used in popular web applications including WordPress, Facebook and Twitter. This data security is essential in our project as we need to protect our user’s data. We will be storing information about the user’s business such as orders and customers. This is important data that must be stored securely.

Although our project will only have a small number of users, MySQL allows it to be expanded to a huge size without putting any strain on the database. It is designed to meet even the most demanding applications while ensuring optimum speed, full-text indexes and unique memory caches for enhanced performance. MySQL also comes with the assurance of 24×7 uptime which makes sure our project will always be accessible.

## 4.2 Backend

When choosing what technologies, we were going to use for our backend we had to take a few things into consideration. We had to make sure that we would be able to create a secure backend capable of connecting to our database and frontend. After researching a few different solutions, we decided to go with the technologies mentioned below.

### 4.2.1 Java

Java is a general-purpose computer-programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. We have worked with Java in every year of our degree and we both feel like it is our strongest language. We felt our skills in the language would give us an opportunity to make a high standard backend. Having used Java as a backend before we both had experience in connecting it to databases and frontends. This knowledge would benefit us greatly when it comes to connecting the separate parts of our project.

After we decided on using Java for the backend, we researched a few ways we could design the backend. We looked at previous work we had done and decided to use the Spring framework. This is discussed in the next section.

### 4.2.2 Spring

The Spring Framework is an application framework and inversion of control container for the Java platform. Spring makes use of Inversion of Control and Dependency Injection to promote good software coding practices and speed up development time. We used Spring Boot which is a part of the Spring framework. Spring Boot is simply defined as a set of pre-configured frameworks and technologies which help to reduce boiler plate configuration. This provided us with a shorter way of getting a basic Spring web application up and running in the quickest amount of time. We used this as it allowed us to focus on other aspects of the project without being bogged down on setting up the configuration of the web application.

## 4.3 Frontend

When choosing what technologies, we were going to use for our frontend we had to take a few things into consideration. We had to make sure that we would be able to create a frontend that could connect to our backend where all the requests would be made. After researching a few different solutions, we narrowed it down to two choices. These were React and Angular. After a more in depth look at both, we decided to choose Angular for our frontend.

### 4.3.1 Angular

Angular is a TypeScript-based open-source web application framework led by the Angular Team at Google and by a community of individuals and corporations. Angular is a complete rewrite from the same team that built AngularJS. Angular allows us to build progressive web applications that use modern web platform capabilities to deliver app-like experiences. They allow our applications to be high performance and to run offline. Building native mobile applications is also possible through Angular with strategies from Cordova, Ionic, or NativeScript. Desktop applications can be created across Mac, Windows, and Linux using the same Angular methods that would be used for web applications.

Speed and performance are key for Angular. Angular can turn your templates into code that's highly optimized for today's JavaScript virtual machines. This gives you all the benefits of hand-written code with the productivity of a framework. Angular code can be rendered nearly instant once the first view of the application is served on Node.js®, .NET, PHP, and other servers. This also paves the way for sites that optimize for SEO.

Angular apps also load very quickly thanks to the new Component Router. This delivers automatic code-splitting, so users only load code required to render the view they request.

Angular also allows for great productivity. Templates can be used to quickly create UI views with simple and powerful template syntax. Angular command line tools allow us to start building fast, add components and tests, then instantly deploy. All popular IDEs and editors have intelligent code completion for Angular, provide instant errors and other feedback for Angular.

All these benefits mentioned above are the reasons we chose Angular to work with. It is in our opinion the best and easiest to pick-up front-end framework.

### 4.3.2 Typescript

TypeScript is an open-source programming language developed and maintained by Microsoft. It is a superset of JavaScript and adds optional static typing to the language. TypeScript is designed for development of large applications. As TypeScript is a superset of JavaScript, existing JavaScript programs are also valid TypeScript programs. TypeScript may be used to develop JavaScript applications for both client-side and server-side execution.

TypeScript supports definition files that can contain type information of existing JavaScript libraries. This enables other programs to use the values defined in the files as if they were statically typed TypeScript entities. There are third-party header files for popular libraries such as jQuery, MongoDB, and D3.js.Node.js programs can also be developed within TypeScript due to TypeScript headers for the Node.js basic modules being available, allowing development of Node.js programs within TypeScript.

Since TypeScript starts from the same syntax and semantics that all JavaScript developers know today, it is possible to use existing JavaScript code, incorporate popular JavaScript libraries, and call TypeScript code from JavaScript. TypeScript compiles to clean, simple JavaScript code which runs on any browser, in Node.js, or in any JavaScript engine that supports at least ECMAScript 3.

TypeScript has tools which enable large application development. Types enable JavaScript developers to use highly-productive development tools and practices. These include like static checking and code refactoring when developing JavaScript applications.

TypeScript is essentially state of the art JavaScript. It offers support for the latest and evolving JavaScript features. These features are available at development time for high-confidence app development but are compiled into simple JavaScript that targets ECMAScript 3 environments.

### 4.3.3 HTML

Hypertext Markup Language is the standard markup language for creating web pages and web applications. Combining HTML with Cascading Style Sheets and JavaScript, you get the triad of cornerstone technologies for the World Wide Web. Web browsers can receive HTML documents from a web server or local storage and render them into multimedia web pages. HTML documents are documents which describe the structure of a web page semantically by use of tags, with these tags representing different actions when the page is rendered.

HTML elements are the building blocks of HTML pages. They are what are used to describe the action taken when the page is rendered. HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a way to create structured documents by using tags to define structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. These tags are written using angle brackets. Tags such as <img /> and <input /> directly bring content into the page. The <img/> tag allows you to give the path to an image which will then be rendered into the web page. Other tags such as <h1> surround and provide information about document text and may include other tags as sub-elements. The <h1> tag shown represents a header. This will cause the text inside the tags to be displayed in a larger size than the text in the rest of the page. Web Browsers do not display the HTML tags, they just use them to interpret the content of the page.

HTML can also use programs written in a scripting language such as JavaScript or TypeScript, which affects the behavior and content of web pages. Cascading Style Sheets define the look and layout of content. HTML is used in our project through the Angular frontend. Angular uses HTML to build the web pages and the content they display. We use the TypeScript in Angular to control certain aspects of the HTML pages.

### 4.3.4 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML.As mentioned above CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. HTML was never intended to contain tags for formatting a web page. HTML was created to describe the content of a web page, such as

<h1>This is a heading</h1>

<p>This is a paragraph</p>

Then tags like <font>, and color attributes were added to the HTML 3.2 specification, it became a nightmare for web developers. Large website development where fonts and color information were added to every single page, became a long and expensive process. Therefore, CSS was designed to allow for the separation of presentation and content, including layout, colors, and fonts. This separation was essential as it improved content accessibility, provided more flexibility and control in the specification of presentation characteristics and enabled multiple web pages to share formatting by specifying the relevant CSS in a separate .css file.

The separation of formatting and content also made it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print or by voice. CSS also has rules for alternate formatting if the content is accessed on a mobile device. While CSS is mainly used with HTML, it is also supported by other markup languages including XHTML, plain XML, SVG, and XUL.

We used CSS in our project inside our Angular frontend to style our HTML elements. It allowed us to create items at certain areas on the page and style them in a way we wanted.



*The cornerstone technology of the World Wide Web*

**4.4 Software Used**

While developing this project we had to use software such as IDEs to write the code for our project. Below we will discuss the software we used while developing our project and for what did we use it for.

**4.4.1 Eclipse IDE**

Eclipse is an integrated development environment and is the most widely used Java IDE. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins.

The Eclipse software development kit, which includes the Java development tools, is meant for Java developers. It is possible to extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages.

We used Eclipse in our development for our backend Spring Boot development. We have used Eclipse for all Java development we have done in the past and this was our reason for using it for this project.

**4.4.2 Visual Studio Code**

Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is also customizable, so users can change the editor's theme, keyboard shortcuts, and preferences.

Visual Studio Code is based on Electron, a framework which is used to deploy Node.js applications for the desktop running on the Blink layout engine. Although it uses the Electron framework, the software does not use Atom and instead employs the same editor component used in Azure DevOps.

We used Visual Studio Code to develop the Angular frontend of our project. We used the TypeScript plugin for Visual Studio Code to give us the full tools available to us while developing. We both had past experiences using Visual Studio Code, therefore we decided it was the best code editor for us to use.

**4.4.2 WAMP**

Windows, Apache, MySQL, and PHP (WAMP) is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP). It is often used for web development and internal testing; however, it can also be used to serve live websites.

WAMP also includes MySQL and PHP. These are two of the most common technologies used in creating dynamic websites. MySQL is a high-speed database, which we use in our project, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can build and test a dynamic website before publishing it to a public web server.

We used WAMP to develop and control our MySQL database. Our database stores all information from our web application. Using WAMP it was possible for us to monitor the database and see when items were being added/deleted from it. This was useful in development as it allowed us to see when requests were being made correctly to and from the database.

**Chapter 5**

# System Design

As many pages as needed.

• Architecture, UML etc. An overview of the different components of the system. Diagrams etc... Screen shots etc.

**Chapter 6**

# System Evaluation

As many pages as needed.

* Prove that your software is robust. How? Testing etc.
* Use performance benchmarks (space and time) if algorithmic.
* Measure the outcomes / outputs of your system / software against the objectives from the Introduction.
* Highlight any limitations or opportuni-ties in your approach or technologies used.

**Chapter 7**

# Conclusion

About three pages.

* Briefly summarise your context and ob-jectives (a few lines).
* Highlight your findings from the evalua-tion section / chapter and any opportuni-ties identified.

**Bibliography**

[1] A. Einstein, “Zur Elektrodynamik bewegter Ko¨rper. (German) [On the electrodynamics of moving bodies],” *Annalen der Physik*, vol. 322, no. 10, pp. 891–921, 1905.