

TAK Computers Online Inventory Management Website

Thomson Ly, Aleksandar Milosevski, Kellen Leclair

*School of Computer Science
University of Windsor
Windsor, Canada
{lyt, milosev2, leclair6}@uwindsor.ca*

Abstract

Our team developed TAK Computers, an online inventory management system specializing in computer hardware. Using the Python-based Django framework and elements of HTML and CSS from COMP-3340, we created a user-friendly, fully functional website. Doing in-depth research, both individually and collectively, was instrumental to our success, enabling us to avoid typical errors and incorporate industry best practices. Our collaborative effort was optimized using Github, ensuring proper coordination and version control with the repository link found in Appendix A. The result is a system with an intuitive interface that scales easily with increasing data. The website's setup facilitates easy integration of new features and provides room for future improvements. TAK Computers not only serves as an efficient inventory management system but is also designed with foresight. We plan to evolve it into a website with full online shopping capabilities, complete with advanced search and shopping integration. This transition aims to transform TAK from an inventory system into a comprehensive e-commerce solution for any company in its field.

I. Introduction

In the 21st century, where business operations are driven by technology and data, dealing with inventory management in an effective way is crucial for businesses to thrive. Online inventory management systems have emerged as a solution to traditional methods of inventory management, which are often inaccurate, manual, and harder to maintain. Online inventory management systems have become an important tool because of their real-time data accessibility, general ease of use, and advanced analytical capabilities. Today more than ever, the world relies on computer hardware parts for everything from small personal devices like phones to huge, powerful data centres. Our TAK Computers online inventory management system was made to streamline the process of adding, organizing, and tracking computer-hardware parts. More specifically, the goal of our project was to create an online inventory management system for a computer hardware parts seller, TAK Computers, that provides a more efficient and effective way of managing inventory than traditional methods. This was achieved with our online system by including many features such as user registration and authentication, adding new items, editing items, and searching/filtering items. There were multiple reasons why we were motivated to create this online inventory management system. First, traditional inventory management is error-prone, outdated, hard to track, and generally does not provide real-time insights. In creating this system, we aim to simplify the otherwise complicated inventory tracking process by making a simple, easy-to-use website. This type of online system is important because it has become crucial for businesses to keep up with technology and all the ways it can help a business. With our online inventory management system, TAK Computers will be able to compete with the top hardware sellers in the market due to improved inventory capabilities and customer satisfaction.

II. Relevant Literature Review

The development and implementation of our online inventory management website was not an isolated effort, and we collected information from various resources to guide us along the way. During our literature review we identified 10 articles from a mix of classical and modern fields that would help us in the development and structuring of our project. Here we will discuss the three articles that ended up being the most important and relevant to our project, by going through each article's strengths, weaknesses, and best practices for our project.

Article 1: Design and Development of an Online Assets Inventory Management System (AIMS) [3]

This paper explores the design, development, and benefits of online asset management. It offers insights into the advantages of automated inventory systems over traditional manual methods. For creating their inventory management system, the methodology used in the paper was a top-down software development approach. The front-end provides an interactive interface for the user to input database information. As for the back end (database), it provides a detailed record of all the assets in the organization. Finally, the web-based system is usable from any terminal with internet access, allowing a user to update assets' details and add new locations/offices to the database.

Strengths:

- **Automation Over Manual Methods:** The AIMS paper emphasizes the importance of online inventory management, highlighting its efficiency, accuracy, and convenience when contrasted with manual spreadsheet-based systems.
- **Universal Accessibility:** The study's focal point on a web-based inventory system ensures users can access and manage inventory from any device, anywhere, provided they have internet access. This promotes flexibility and broadens user reach.

- **Technological Integration:** Employing technologies such as PHP, HTML, MySQL, and CSS, the AIMS system provides a comprehensive online platform that's interactive, responsive, and user-friendly.

Weaknesses:

- **Limited to Asset Management:** While the AIMS paper focuses on asset management, it does not necessarily branch out to cater to specific niches like computer-related items, which is a primary focus for TAK Computers.
- **Technological Dependence:** Their reliance on specific technologies, such as the Wamp Server, might restrict adaptability or integration with other platforms or updates in the future.

Best Practices and Application to Our Project:

The best practices from this article that were relevant to the implementation of our website were the focus on real-time collaboration of AIMS and universal accessibility aspects. We implemented this in our workflow by employing Git/Github for cohesive real-time editing. The universal accessibility aspect emphasized in the paper was important for us, and we prioritized this by creating virtual environments to run exactly the same on all operating systems. Integration of advanced technologies and automation was another important part of the paper. We tied this into our project by using a Django framework, enhancing the field of automated over manual methodology.

Article 2: An Online Based Inventory Management System Implementation in Printing Business [2]

This article provides an in-depth examination of an online inventory management system tailored for the printing business. By focusing on the challenges of manual labour, extensive paperwork, and recurring human errors in conventional systems, it introduces an innovative solution using digitalization and automation. The core benefits encompass enhanced product categorization, increased stock visibility, and the automation of invoice creation.

Strengths:

- **Real World Implementation:** An important feature of this study is its presentation of a real world application. This wasn't just an academic exercise; it has been applied and used in an actual business context, offering readers a concrete perspective.
- **Operational Enhancement:** The article doesn't stop at introducing digital solutions. It takes a holistic approach to enhancing multiple facets of the business operation, streamlining processes, and ensuring efficiency.

Weaknesses:

- **Limited Client Interaction:** A noticeable gap is the lack of a client-side ordering system. Such a feature would have increased the overall functionality and user experience of the system.

Best Practices and Application to Our Project:

This article serves as a demonstration of how an online inventory system can be tailored to address specific industry challenges. By breaking down its implementation, we

gained valuable insights that helped us to navigate our project's direction. The emphasis on real-world application acted as a continual reminder of the challenges and importance while creating an inventory management system to solve practical problems. This assisted us in focusing more on the applied aspects of the project rather than strictly the academic.

Article 3: Development of a Web-based Platform for Automating an Inventory Management of a Small and Medium Enterprise [1]

This article delves into the challenges associated with traditional inventory management systems, highlighting issues such as their labour-intensive nature and a higher propensity for errors. In response, it suggests a digital platform as a solution. The detailed descriptions of the system's features, from database connectivity (using MySQL) to various user interfaces like the login page, sales report page, dashboard, and more, give us practical insights into the development process. It has a focus on small and medium enterprises, which coincides with the scope of our project's initial implementation. With included visuals, it provided a comprehensive view that aligned with our project's aim to build an efficient inventory management system.

Strengths:

- **Detailed Implementation:** The article excels in illustrating the intricacies of designing and developing an inventory management system. By using the iterative waterfall software development model, it ensures errors are detected and resolved early.
- **Comprehensive Overview:** The extensive presentation of user interfaces, coupled with their functionalities, provides a clear blueprint of what a functional digital inventory system looks like.

Weaknesses:

- **Lack of Limitations/Future Work:** The article doesn't address potential shortcomings or future improvements, leaving a gap in understanding the system's adaptability or scalability.
- **Incomplete Technical Details:** While the article delves deep into certain technical aspects, it falls short in detailing others, such as the framework employed, which would have offered a more complete perspective.

Best Practices and Application to Our Project:

The article's detailed portrayal of a digital inventory management system offers us valuable insights. Its methodical approach, particularly with the iterative waterfall model, reinforces the importance of proactive error detection and resolution. The wide array of interfaces, backed by visuals, provides an easily understandable blueprint, which guided us in designing our own system. Emulating such a comprehensive approach, while also addressing the article's gaps, assisted in making sure our project was functional and helped us to achieve our goals while avoiding noted pitfalls.

III. Project Details and Methodology

To create our TAK Computers online inventory management system, we used a combination of Django, HTML, and CSS. We started first by learning about Django and how it can be implemented. Django aligns with the development of an online inventory management system and was responsible for the back-end development. Utilizing Django allowed us to easily set up databases as Django has built-in Object-Relational Mapping (ORM), which lets us manage the database operations easily. Also, with built-in security protections, Django shields us from many common security concerns and keeps data entered into our system safely.

HTML was important in the development of our system because it gave the website foundational structure. Using HTML, we were able to add many simple, yet crucial features such as product listing, description, image, and more. This was important in offering users of the site a structured view of items in the system as well as an intuitive and user-friendly interface. Integrating HTML with Django, we were able to also implement many other important features like adding items and user registration/authentication.

CSS was also used in the development of our system and served as the main instrument for all things design and appearance related. Using CSS, we were able to make the website more aesthetically pleasing by doing things like changing fonts, colours (background, font, buttons, etc.) and spacing. CSS also played an important role in making a responsive design that would make our website work across different devices and ensured that user experience is friendly by having a good looking, easy to navigate website.

By leveraging the power of Django, HTML, and CSS, we created a modern, simple, and effective online inventory management system for computer-hardware items as a solution to outdated traditional methods. This system was not only visually appealing and easy to use, but it was also fully functional and safe.

IV. Definitions

The following definitions will help provide clarity and may help to better understand some specific terms in our project:

- **Front-end development:** the process of making the visual and interactive parts of a website/application that a user directly interacts with (text, images, buttons, etc.) The technologies used in this project for front-end development are HTML and CSS.
- **Back-end framework:** sets of tools and libraries that help developers in building the server-side of websites/applications. This part of development is not directly visible to end-users, and it deals with tasks like database interactions, security, etc. The back-end framework used in this project was Django.
- **Online inventory management system:** a web-based solution that allows individuals or businesses to track and manage their products in real time. Allows users to access and monitor inventory data from any device with an Internet connection.
- **Object-Relational Mapping (ORM):** An approach in programming that bridges the gap between object-

oriented programming languages and relational databases. The ORM technique facilitates the seamless conversion of data between these differing (incompatible) systems. In the context of Django, its ORM simplifies database interactions by enabling developers to work with databases using Python objects

V. Specification/Features

Gathering inspiration from features commonly found in online inventory management systems across the web, the key functionalities to provide users with a comprehensive inventory management website are discussed in the following section. Image snapshots of the described sections for visualization of the website can be seen in Appendix B.

Front Page

Starting at the top of the front page, the top left corner is the logo and name of the website, TAK Computers, which if clicked on, will redirect the user to the front page, regardless of the page that the user is on. The top right corner of the page contains interactive elements that differ depending on if the user is logged in or not and can consist of the following: Add New Item, Browse, Sign up, Login, Dashboard, Profile, and Logout. The middle of the page displays six of the various available items in TAK Computers inventory, specifically each item's image, name, and price. Clicking on any of the items would redirect the user to the item's detail page. Below the six displayed items is a pagination feature that allows the user to scroll through more pages of available items. This pagination feature is also implemented in the Browse page and Dashboard page discussed later. After the available items are the computer categories section which shows all the different computer categories available on TAK Computers (e.g., CPU, case, GPU). Clicking on any of these categories will bring you to the Browse page with the filter selected for that clicked category. At the very bottom of the front page is an "About" TAK Computers section on the left, and a clickable "Menu" on the right with the options: Front Page, About Tak Computers, Contact Us, and Privacy Policy, which redirects to each of those specific pages.

User Registration, Authentication and User Logout

On any page of the website, at the top right corner, there are both the "Sign up" and "Login" buttons. When clicking on the Sign up button, the user is redirected to a signup form that requests the following: a username of any length, a valid email, a password, and a repeat password of at least 8 characters. After successful registration, the user will be redirected to the login page which requests for a valid username and corresponding password in the database (Note: these two fields are case-sensitive). After successful user authentication, other features that require the user to be logged in are now available. Specifically, the two previous buttons, signup, and login, are now replaced with the "Profile" and "Logout" buttons respectively, along with an additional button to the left of profile, which is the "Dashboard" button. Clicking on the Logout button will successfully logout the user and a dismissible alert indicating proper logout has occurred will be displayed.

Adding Items and Item Details Page

When adding a new item to TAK Computers inventory, it is required that the user must be logged in first. Thus, when clicking on the “Add New Item” button at the top of the page, if the user is not currently logged in, they will be redirected to the login page. Otherwise, an authenticated user is redirected to an add item form with the following sections to be filled out: computer-hardware category, name, description, price and image of the designated item to be added. After successfully filling out the form and clicking submit, the user will then be redirected to that specific item’s detail page. On this page, found on the left will be the item’s image, while on the right will be a details box with the item’s name, price, lister (e.g., current user) and description. As well, there are two additional buttons present depending on if the lister is the current user, which is an “Edit Item” and “Delete Item” button. Additionally, at the bottom of the page, there will be “Related Items” displayed which showcases items of the same category as the current item. These are interactable items that will lead the user to that designated item’s detail page. Clicking on the edit item button redirects the user to an almost identical add item form, but with an additional checkbox this time for “Is sold”, which can be clicked if the item is sold and not in the inventory. Changing an item to “Is sold” will have the item not be present on the website’s front page and browse section, but instead appear only in the user’s dashboard.

Browsing Items

Clicking on “Browse” at the top redirects you to the browse page which has searching/filtering on the left, and items on the right. Starting with the search box and button, inputting keywords, and clicking the search button will scan through items that contain that specific keyword in the item’s name or description. For the filters, clicking on any of the categories displayed such as CPU, will filter and display items only of that category. Like the front page, clicking on any of the displayed items on the right will redirect the user to that specific item’s details page.

User Dashboard

As mentioned above, the Dashboard button only appears if there is an authenticated user signed into the website. Clicking on such a button will bring you to the user’s dashboard which displays the specific items listed by the current user (up to six per page). Clicking on any of the items will bring you to the item’s details page like the front page or browsing section. The purpose of the dashboard is to provide the user a clean and simple interface that allows them to organize, edit, and delete any of their listed items, instead of searching through the front page of browse section, which may contain items listed by other users.

Profile and Editing Profile

Like the Dashboard button, the Profile button will only appear if the user is logged in. Clicking on this button will show the user’s profile which consists of their name, email, address, phone number, and a profile picture (default if none is added). There is an additional “Edit” button that allows the user to easily edit such details through the profile edit page form. Note that if a user changes their name through

this edit button, their corresponding username for logging in will now use that new edited username.

VI. Platform

Our project, TAK Computers, was developed on the Django web framework, along with the mentioned combination of HTML and CSS for the front-end development. The reason we chose this platform was because Django is a high-level, open-source framework written in the simple and straightforward language of Python. Python’s simplicity and readability, combined with the well-documented nature of Django, alongside a plethora of online resources, allowed us, as first-time programmers in Django and beginners in Python, to rapidly acquaint ourselves with the framework’s intricacies. Django provides the perfect foundation for a complete web application like this online inventory management system project.

A close alternative would have been Flask, but Django was chosen as the optimal choice due to its lower learning curve, as a good example is the implementation of the user registration and authentication system. This implementation is much simpler in Django as it has user management built into the framework through the mentioned ORM system while in frameworks like Flask, manual creation of databases using SQL is instead required.

An incredible advantage of Django during development was the use of its template system for HTML. Having a base template that is inherited across different HTML files allowed for the simple implementation of the different web pages across the website. Django templates thus allowed us to have a consistent, coherent, intuitive, and user-friendly interface. As well, any changes to the base template would be propagated to all extended HTML files, thus reducing the likelihood of page discrepancies and errors. Alongside this, Django templates enable us to have variables, loops and conditional statements directly in the HTML files which made it easy to implement features like pagination seen on the front, browse and dashboard pages in only a couple of lines.

Overall, the Django platform serves as the ideal framework for our project’s online inventory management system due to its user-friendly nature, robust back-end capabilities, rapid development capabilities and excellent integration with HTML and CSS. The framework’s efficiency and simplicity in implementing a full-fledged web application align perfectly with our goals of creating a complete online web inventory system.

VII. Design

Present the design of your project, referring to the online shop web app example. Discuss the importance of a clean and intuitive user interface, showcasing clear navigation menus, prominent product images, and streamlined checkout processes.

In the design of our TAK Computers online inventory management system, we thoughtfully developed a user interface that prioritizes clarity and intuitiveness, drawing inspiration from clean designs found on similar websites. The visual design of the TAK Computers website can be seen

through the screenshots shown in Appendix B. The approach of our design ensures that users can seamlessly navigate through the website due to the consistent header and footer found from page to page. As noted in the photos shown in Appendix B, the header consists of the website logo and name on the left which allows a user to always have a home button to the front page, regardless of where they are within the website. Comparatively, the right side of the header contains essential interactive elements to the inventory management system, which consists of options to add items, redirection to the browse items page, and user-related features like registration/authentication, dashboard, and profile. The strategic placement of these essential features in the header enhances user experience as it becomes a functional hub that offers immediate access to key functionalities from anywhere within the website. It allows users to perform crucial actions (e.g., adding an item to the inventory) without the need to navigate extensively through the website's pages.

Alongside this, simple but crucial checks within the website design, such as requiring a user to be logged in first in order to add an item, ensure a secure and controlled environment for the addition of items to the inventory. This prevents unauthorized users from submitting information to the database and maintains the integrity of our website's data. As well, redirecting an unauthorized user to the login page when clicking on the "Add New Item" element streamlines the process. This makes the process more user-friendly, as it indicates to the user to either log in as an existing user or to use the signup button seen in the header to create a new user first.

The development of an intuitive user interface continues for the remaining features of the website. Clicking on an item on the front page, browse section or dashboard will bring you to that item's detail page. Within this details page, additional related items are also presented, enabling users to have quick and easy access to these items with a single click. This approach extends to category navigation, where choosing a category on the front page brings you directly to the browse page with that specific category filter selected. These design choices minimize any unnecessary steps for users, allowing them to immediately have access to items or pages of interest without searching extensively.

Overall, our user-centric design focuses on creating an effortless and efficient experience for users, with the goal of streamlining inventory management. Through considerate placement of essential features to intuitive navigation pathways, TAK Computers online inventory management system becomes an invaluable resource, offering a user-friendly experience to visitors of our platform.

VIII. Sample Implementation Details

A sample implementation detail of a crucial feature in TAK Computers is the development of the "Add Item" feature. This required many major steps like HTML and CSS files for the front-end, and Django Models, Forms, Views, and URLs for the back end. This is a simple yet important feature for any online inventory management system to have. Code snippets for the following mentioned files are shown in Appendix C. To implement this feature, first, we created an HTML page called `form.html` which inherits from our `base.html` Django template. This HTML page serves as a form

for filling out item information and includes basic formatting as well as checks for form errors before allowing a user to submit. Afterwards, within the back end, a function was implemented for adding a new item. Initially, we must create a model for categories and items in `models.py` which includes all of their fields and corresponding value types. The next step is to create a "NewItemForm" class in `forms.py`, which will help in simplifying data submission and data validation. The NewItemForm class has the fields we expect an item to have (name, category, description, etc.) along with the styling we would want the form to appear in. Afterwards in `views.py`, in order to add a new item, we make a "new" function. This function takes a 'request' object which holds information about the current web request. It checks if the request.method is 'POST' and if so, a new instance of the NewItemForm is created. The function is also responsible for checking to make sure the information entered is valid, and if so, the item is then assigned to whichever user created it, after which it is saved to the database. If all was successful, the user is directed to the new detail page of the item they entered. The final step is creating a path in the `urls.py` file for the new function which is seen on line 10 within the "urlpatterns" list.

This was an example of how implementation was done for a single feature of our online inventory management system. The integration of the front-end with the back end was a fairly seamless task. The implementation of other features such as editing or deleting an item was very similar to the implementation of adding an item described above.

IX. Discussion

We learned a lot both individually and as a group while working on this project. Although having a larger project with some freedom to be creative was a nice change from what we've been involved in up to this point, we also found that it came with many challenges. The initial problem we encountered was coming up with a base idea and framework for how to approach our project. We had to communicate well as a group and narrow in on our interests for what we'd like to implement. Given our minimal coding experience with larger projects, we aimed to select a website option that was manageable and straightforward to execute, especially in the initial stages.

Firstly, we decided that we would tackle the problem of creating an online inventory website. It seemed like the most clear-cut example given by the professor and there were many resources that we could reference to help us get started and model our project. This straightforward starting point, combined with the opportunities to expand and add more complex features as we progressed seemed like a great option for our level of experience. By starting with a strong literature review and finding relevant articles to support our project, we were able to get a good understanding of what it would take to develop a well-structured inventory management website. We studied all the articles in depth and by the time we initiated our project, we had a knowledgeable perspective of what it would take to develop a useful inventory management system in a real-world setting. We also were able to identify pitfalls and complications to look out for which assisted greatly in setting the goals and functionalities that we wanted to achieve.

When it came to implementing the code for our website, we started by first looking at different frameworks that could be suitable for our project. This aspect of preparation was one of our main strengths and themes throughout the project. We decided that coding in Python using a Django framework would be a great starting point for the coding, due to the plethora of resources and tutorials for designing the back end of a website with this structure. Due to this and our learning of HTML and CSS for the front end coding, much of our difficulty at this point came not from the actual coding itself, but from the communication and teamwork required to create a cohesive website.

The way we dealt with this challenge was by creating a Github repository and learning how to effectively use Github and Git to make all of our changes in real time and see what each person was working on. In this way, we were able to see and edit one another's changes and delegate workflow effectively. We combined this with an active messaging group with strong participation from each member. This part may seem trivial, but it was an extremely important aspect where we talked daily, came up with implementation ideas, and learned from each other by constantly summarizing what each member was learning and working on. We were able to communicate effectively on bugs we encountered while coding, and any difficulties while using Git and setting up virtual environments to work on code individually. This drastically increased the speed of our problem-solving and overcoming obstacles throughout our project.

Our team's dynamics, especially how we divided the tasks, played a massive role in shaping our project. Each of us had our own niche areas to handle, but looking back, we would have benefited from more joint brainstorming and feedback sessions. As our project began to branch out in complexity, the need for deeper teamwork was clear. It might've been better to mix our skills more instead of sticking solely to one person's expertise for certain parts. This way, we'd have a clearer grasp of the whole picture, and it would've made adding new features smoother.

X. CONCLUSION AND FUTURE WORK

Developing the online inventory website brought forth various challenges and learning opportunities. Key obstacles such as mastering Django, Python, Git/Github, and navigating virtual environments were successfully overcome due to our excellent ability to adapt and strong foundation in preparation, collaboration, and communication. One lesson learned was the importance of broader collaboration across different project sections. As the project grew in complexity, the need for a more integrated approach became evident, suggesting that future projects would benefit from a holistic involvement of all team members across all facets. Overall, each group member was able to contribute significantly and advance our proficiency in this field, and the outcome of the project was up to par with what we expected and hoped to achieve.

As we reflect on the success of our inventory management website project, we have also been able to identify areas of expansion and features which would improve the website in the further stages of taking it online to a customer base and implementing it in a real-world setting. Some of these are based on more technical aspects of

improvement, while others are more geared toward customer satisfaction and business/sales-related optimization. Some of the main features we've identified as important in future work are as follows:

1. Online hosting: Our initial design was based on a local hosting environment, which served well for the prototyping and testing phase. Our next step is to migrate this to an online server.
2. Password Recovery: Once the site is online and has more users, it is inevitable that people will forget their passwords. Implementing an automatic recovery system (e.g., via email) will remedy a common problem which is dealt with on all professional websites of this type.
3. Image Carousel/Gallery: As it stands, we have only one photo per item. The importance of a product's visual representation cannot be overstated, and adding an image carousel for each item would allow users to get a comprehensive view of each product.
4. Responsive Design: Another visual-related aspect is optimizing responsive design. By making sure the website presentation is clean and consistent on common platforms such as mobile phones, tablets, and various size laptops and desktop monitors, it ensures maximal user engagement across all platforms.
5. Advanced Search and Filtering: With an expanded inventory that would be evident in a large company, it is imperative that we improve search and filtering methods on our website to assist the user in finding their ideal products. This includes searching based on multiple categories, brands, and selected price ranges.
6. Expansion into Online Shopping: Building on advanced search and filtering, the next step for our website would be to expand and implement online shopping so that the website is both inventory management-based and shopping based. This would increase complexity much further as many other vital features would need to be implemented like the ability to save multiple items in a cart as they go. This way, the user can add various items and see how it stacks up in their budget. Also, for customers looking to make a setup with multiple components, it will allow them to see various combinations and analyze the software, hardware, and visual/spacing related components of their overall setup.

REFERENCES

- [1] Agboola, F. F., Malgwi, Y. M., Mahmud, M. A., and Oguntoye, J. P. (2022). Development of a Web-Based Platform for Automating an Inventory Management of a Small and Medium Enterprise. *FUDMA Journal of Sciences*, vol. 6, no. 5, pp. 57-65. <https://doi.org/10.33003/fjs-2022-0605-1064>
- [2] Ara, R., & Rahim, M. A. (2018). An Online Based Inventory Management System Implementation in Printing Business. *Journal of Emerging Technologies and Innovative Research*, vol 5, no. 11. <https://www.jetir.org/papers/JETIR1811B29.pdf>
- [3] Ayangbekun Oluwaemi, J. & Adele-Sanyaolu Ibrahim, A. (2014). Design and Development of an Online Assets Inventory Management System


(AIMS). International Journal of Scientific Research and Management, vol. 2, no. 8, pp. 1187-1192.
<https://ijsrm.net/index.php/ijsrm/article/view/845/742>

APPENDIX A

Github repository: <https://github.com/Thomshu/TAK-Computers-Website>


APPENDIX B

Front Page



TAK Computers

[Add New Item](#)
[Browse](#)
[Dashboard](#)
[Profile](#)
[Logout](#)


Available Items




AMD Ryzen 5 5600 CPU
Price: \$183.00




AMD Ryzen 5 7600X CPU
Price: \$319.00




Dell 27 Gaming Monitor
Price: \$399.99



MSI Gaming Radeon RX 6650 XT GPU
Price: \$309.99



G.SKILL Ripjaws V Series 32GB RAM
Price: \$99.97



Patriot P210 SATA 3 2TB
Price: \$99.99

[1](#)
[2](#)
[3](#)
[4](#)
[5](#)
[Next](#)
[Last](#)

Computer Categories


CPU 4 Items total	Case 2 Items total	GPU 4 Items total
Hard Drive (HDD) 2 Items total	Monitor 2 Items total	Motherboard 4 Items total
PSU (Power Supply) 4 Items total	RAM (Memory) 4 Items total	SSD 4 Items total

[About](#)

[Front Page](#)
[About TAK Computers](#)
[Contact Us](#)
[Privacy Policy](#)

Based in Windsor, Ontario, our inventory management website is designed to efficiently meet the high demands of our business.

User Registration, Authentication and User Logout

 **TAK Computers**

[Add New Item](#) [Browse](#) [Sign up](#) [Login](#)

Sign Up


Username

Email

Password

Verify Password

[Submit](#)

 **TAK Computers**


[Add New Item](#) [Browse](#) [Sign up](#) [Login](#)

Login User

Username


Password

[Submit](#)


 **TAK Computers**

[Add New Item](#) [Browse](#) [Dashboard](#) [Profile](#) [Logout](#)


Available Items



AMD Ryzen 5 5600 CPU
Price: \$183.00




AMD Ryzen 5 7600X CPU
Price: \$319.00



Dell 27 Gaming Monitor
Price: \$399.99

Adding Items and Item Details Page


TAK Computers

[Add New Item](#)
[Browse](#)
[Dashboard](#)
[Profile](#)
[Logout](#)

Add New Item

Category:

Case

Name:

Cooler Master NR200 White

Description:

Sample Description


Price:

1299.99

Image:

[Choose File](#)
1 (DEMO) Cooler Master NR200 White CASE.jpg

Submit



Cooler Master NR200 White

Price: \$1299.99


Lister: Thomson

Description:
Sample Description

You own this item


[Edit Item](#)
[Delete Item](#)

Related items



NZXT H5 Flow Compact ATX Mid-Tower


Price: \$109.99



Lian Li LI PC-O11 Dynamic EVO Black Case

Price: \$199.99

Browsing Items via Keyword Search and Filters



TAK Computers

[Add New Item](#)
[Browse](#)
[Dashboard](#)
[Profile](#)
[Logout](#)


Gaming

Categories


- CPU
- Case
- GPU
- Hard Drive (HDD)
- Monitor
- Motherboard
- PSU (Power Supply)
- RAM (Memory)
- SSD




Dell 27 Gaming Monitor
Price: \$399.99




MSI Gaming Radeon RX 6650 XT GPU
Price: \$308.98




MSI RX 7900 XTX GAMING TRIO CLASSIC 24G GPU
Price: \$1249.00




Alienware 38 Curved Gaming Monitor
Price: \$1296.99



MSI MPG Z790 Edge WiFi Gaming Motherboard
Price: \$368.01



Lian Li Li PC-O11 Dynamic EVO Black Case
Price: \$199.99



TAK Computers

[Add New Item](#)
[Browse](#)
[Dashboard](#)
[Profile](#)
[Logout](#)


Search for RAM, CPU, GPU, MOBO...

Categories


- CPU
- Case
- GPU
- Hard Drive (HDD)
- Monitor
- Motherboard
- PSU (Power Supply)
- RAM (Memory)**
- SSD




G.SKILL Ripjaws V Series 32GB RAM
Price: \$89.97



TEAMGROUP T-Create 64 GB RAM (2x32 GB)
Price: \$109.99



CORSAIR Vengeance RGB 32 GB (2x16GB) DDR5 RAM
Price: \$339.99









Patriot Viper Venom DDR5 32GB (2 x 16GB) RAM
Price: \$165.00

User Dashboard

TAK Computers Add New Item Browse Dashboard Profile Logout

Thomson's Listed Items

 <p>AMD Ryzen 5 7600X CPU Price: \$179.00</p>	 <p>G.SKILL Ripjaws V Series 32GB RAM Price: \$80.00</p>	 <p>Patriot P210 SATA 3 2TB Price: \$100.00</p>
		


Profile and Editing Profile

TAK Computers Add New Item Browse Dashboard Profile Logout

User Profile Page

Profile Information Edit

Name	Thomson
Email	thomson.jy98@gmail.com
Address	869 Bruce Avenue
Phone	5199889763



TAK Computers Add New Item Browse Dashboard Profile Logout

Profile Edit Page

Username*


Email*

Address*

Phone*

Image
 Currently: Profile_Images/cropped_prof_pic_ETSDONC.png ☐
 Clear
 Change: No file chosen

About TAK Computers, Contact Us and Privacy Policy Page



TAK Computers

[Add New Item](#)
[Browse](#)
[Dashboard](#)
[Profile](#)
[Logout](#)


About TAK Computers

TAK Computers is a computer hardware parts seller based in Windsor, Ontario. Our goal is to provide customers the most modern, high quality computer hardware parts to meet a wide variety of needs.

We have developed an inventory management website to meet the fast-growing demands from customers. This system allows us to add and categorize products for sale and browse products based on category. Each product has a description with a price, and upon selecting a particular product to view, the website will also show you similar products. Products can be added and removed with ease, making this an efficient yet easy to use website for inventory management.




Windsor, Ontario, Canada



TAK Computers

[Add New Item](#)
[Browse](#)
[Dashboard](#)
[Profile](#)
[Logout](#)


Contact Us: TAK (T)homson (A)leksandar (K)ellen!




Thomson Ly
ly@uwindsor.ca | 519-988-9763



Aleksandar Milosevski
milosev2@uwindsor.ca | 226-344-6497



Kellen Leclair
leclair6@uwindsor.ca | 226-346-9432


TAK Computers

[Add New Item](#)
[Browse](#)
[Dashboard](#)
[Profile](#)
[Logout](#)

TAK Computers Privacy Policy

Our privacy policy ensures that all personal information provided by users on our website is securely stored and used solely for the purpose of improving user experience and providing relevant services. We are committed to safeguarding your data and will never share it with third parties without your explicit consent.

At TAK Computers we know how important it is to protect your personal information. When you make an account on the website, your information is safely stored in a secure database. We collect information you give to us such as your:

- Name
- Email
- Phone Number
- Address

We do not share, sell, or rent your personal information to any third parties, except as required by law or with your explicit consent. Your data may be used for internal analysis, system maintenance, and to enhance your user experience. We utilize industry-standard security measures to protect your information from unauthorized access, disclosure, or alteration. By using our website, you consent to the terms of this Privacy Policy. If you have any questions or concerns regarding your data or this policy, please contact us. Contact information can be found on the 'Contact Us' page.

APPENDIX C

form.html

```

1 {% extends 'core/base.html' %}
2
3 {% block title %}{{ title }}{% endblock %} <!-- this 'title' refers to the title "New Item" in views.py -->
4
5 {% block content %}
6 <h1 class="mb-6 text-3xl white-text">{{ title }}</h1>
7 <form method="post" action="." enctype="multipart/form-data">
8     {% csrf_token %}
9     <div class="mt-0 px-6 py-6 bg-gray-100 rounded-xl">
10         <div class="space-y-4">
11             {{ form.as_p }}
12         </div>
13
14         {% if form.errors or form.non_field_errors %}
15         <div class="mb-3 p-6 bg-red-100 rounded-xl">
16             {% for field in form %}
17                 {{ field.errors }}
18             {% endfor %}
19             {{ form.non_field_errors }}
20         </div>
21         {% endif %}
22     </div>
23
24     <button class="mt-6 py-3 px-12 text-lg bg-teal-500 hover:bg-teal-700 rounded-xl text-white">Submit</button>
25 </form>
26 {% endblock %}

```

models.py

```

1 from django.contrib.auth.models import User
2 from django.db import models
3
4 # Create your models here.
5 class Category(models.Model):
6     name = models.CharField(max_length=255)
7
8     class Meta:
9         ordering = ('name',)
10         verbose_name_plural = 'Categories'
11
12     def __str__(self):
13         return self.name
14
15
16 class Item(models.Model):
17     category = models.ForeignKey(Category, related_name='items', on_delete=models.CASCADE)
18     name = models.CharField(max_length=255)
19     description = models.TextField(blank=True, null=True)
20     price = models.FloatField()
21     image = models.ImageField(upload_to='item_images')
22     is_sold = models.BooleanField(default=False)
23     created_by = models.ForeignKey(User, related_name='items', on_delete=models.CASCADE)
24     created_at = models.DateTimeField(auto_now_add=True)
25
26     def __str__(self):
27         return self.name

```

forms.py

```

1  from django import forms
2  from .models import Item
3
4  classNewItemForm(forms.ModelForm):
5      class Meta:
6          model = Item
7          fields = ('category', 'name', 'description', 'price', 'image')
8
9          widgets = {
10              'category': forms.Select(attrs={
11                  'class': 'w-full py-4 px-6 rounded-xl border'
12              }),
13              'name': forms.TextInput(attrs={
14                  'class': 'w-full py-4 px-6 rounded-xl border'
15              }),
16              'description': forms.Textarea(attrs={
17                  'class': 'w-full py-4 px-6 rounded-xl border'
18              }),
19              'price': forms.TextInput(attrs={
20                  'class': 'w-full py-4 px-6 rounded-xl border'
21              }),
22              'image': forms.FileInput(attrs={
23                  'class': 'w-full py-4 px-6 rounded-xl border'
24              })
25          }
26

```

views.py

```

54  @login_required
55  def new(request):
56      if request.method == 'POST': #this is if the form is submitted
57          form =NewItemForm(request.POST, request.FILES)
58
59          if form.is_valid():
60              item = form.save(commit=False)
61              item.created_by = request.user
62              item.save()
63
64              return redirect('item:detail', pk=item.id)
65      else: #Aka if its a get request
66          form =NewItemForm()
67
68      return render(request, 'item/form.html', {
69          'form': form,
70          'title': 'Add New Item', #for forms.html
71      })

```

urls.py

```

3  from django.urls import path
4  from . import views
5
6  app_name = 'item' |
7
8  urlpatterns = [
9      path('', views.browse, name='browse'),
10     path('new/', views.new, name='new'),
11     path('<int:pk>/', views.detail, name='detail'),
12     path('<int:pk>/delete/', views.delete, name='delete'), #for delete items path
13     path('<int:pk>/edit/', views.edit, name='edit'), #for Edit items path
14 ]

```