**ASSIGNMENT 2 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 16: Cloud Computing | | |
| **Submission date** | 30 October 2022 | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** | Truong Van Tuan Kiet | **Student ID** | GCC200203 |
| **Class** | GCC0903 | **Assessor name** | Le Huy Quoc Bao |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice. | | | |
|  |  | **Student’s signature** |  |

**Grading grid**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| P5 | P6 | P7 | P8 | M3 | M4 | D2 | D3 |
|  |  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Internal Verifier’s Comments:** | | |
| **Signature & Date:** | | |

**ASSIGNMENT 2 BRIEF**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number** | Unit 16: Cloud Computing | | |
| **Assignment title** | Cloud’s implementation and security threats | | |
| **Academic Year** | 2021 – 2022 | | |
| **Unit Tutor** | DO Quoc Binh | | |
| **Issue date** |  | **Submission date** |  |
| **IV name and date** |  | | |

|  |
| --- |
| **Submission Format:** |
| *Format:*  A report(in PDF format)  You must use font *Calibri size 12, set number of the pages and use multiple line spacing at 1.3. Margins must be: left: 1.25 cm; right: 1 cm; top: 1 cm and bottom: 1 cm.* The reference follows Harvard referencing system.  *Submission* Students are compulsory to submit the assignment in due date and in a way requested by the Tutors. The form of submission will be a soft copy posted on <http://cms.greenwich.edu.vn/>  *Note:* The Assignment *must* be your own work, and not copied by or from another student or from  books etc. If you use ideas, quotes or data (such as diagrams) from books, journals or other sources, you must reference your sources, using the Harvard style. Make sure that you know how to reference properly, and that understand the guidelines on plagiarism. *If you do not, you definitely get failed* |
| **Unit Learning Outcomes:** |
| **LO3** Develop Cloud Computing solutions using service provider’s frameworks and open source tools.  **LO4** Analyse the technical challenges for cloud applications and assess their risks |
| **Assignment Brief and Guidance:** |
| **Task 1**  Base on the scenario and architecture design in the first assignment provide the implementation. Because of the time constraint of the assignment, the implementation just provides some demo functions of the scenario. The implementation includes two parts:   * A step by step instruction   + which shows which functions are implemented   + How to config, deploy and test the services (Web application, Database Server, Source code management, server logs..) using service provider’s frameworks and open source tools.   + Images for the built functions * A brief discussion about difficulties which one can face during the development process(optional) * The source code for the built application   **Task 2**  The table of contents in your security manual (which should be 500–700 words) should be as follows:   1. Analysis of the most **common problems** and **security issues** of a cloud computing platform. 2. Discussion on how to overcome these issues. 3. Summary. |

|  |  |  |
| --- | --- | --- |
| Learning Outcomes and Assessment Criteria | | |
| Pass | Merit | Distinction |
| **LO3** Develop Cloud Computing solutions using service provider’s frameworks and open source tools | | **D2** Critically discuss how one can overcome these issues and constraints. |
| **P5** Configure a Cloud Computing platform with a cloud service provider’s framework.  **P6** Implement a cloud platform using open source tools. | **M3** Discuss the issues and constraints one can face during the development process. |
| **LO4** Analyse the technical challenges for cloud applications and assess their risks | |  |
| **P7** Analyse the most common problems which arise in a Cloud Computing platform and discuss appropriate solutions to these problems.  **P8** Assess the most common security issues in cloud environments. | **M4** Discuss how to overcome these security issues when building a secure cloud platform. | **D3** Critically discuss how an organisation should protect their data when they migrate to a cloud solution. |

**Table of Contents**

[**Table of Contents** 6](#_Toc118057056)

[**I. Deploy the website.** 9](#_Toc118057057)

[**1. How to deploy (put the website) to Heroku and connect to Postgres** 9](#_Toc118057058)

[**2. Provide pictures of implemented functions** 20](#_Toc118057059)

[2.1. Over view 20](#_Toc118057060)

[2.2. Index 21](#_Toc118057061)

[2.3. Registration page 22](#_Toc118057062)

[2.4. Login 23](#_Toc118057063)

[2.4. List of products 23](#_Toc118057064)

[2.6. Product detail 24](#_Toc118057065)

[2.7. Cart page 24](#_Toc118057066)

[2.8. Create order 25](#_Toc118057067)

[2.9. Order history 26](#_Toc118057068)

[2.10. Manegement product 26](#_Toc118057069)

[2.11. Manegement suplier. 29](#_Toc118057070)

[2.12. Manegement shop. 30](#_Toc118057071)

[2.13. Manegement Order. 32](#_Toc118057072)

[2.14. Manegement shop. 33](#_Toc118057073)

[2.15. Manegemrnt account 36](#_Toc118057074)

[**3. Provide my own difficulties while developing cloud solution** 36](#_Toc118057075)

[**II. Security** 37](#_Toc118057076)

[**1. Discuss common cloud issues and security issues** 37](#_Toc118057077)

[1.1 Consider Basic Security Risks 37](#_Toc118057078)

[1.2. Considerations about Data Security 39](#_Toc118057079)

[**2. Provide solutions of some issues which I can** 40](#_Toc118057080)

[2.1. Countermeasures for Security Risks 40](#_Toc118057081)

[2.1. Methods to ensure Data security 41](#_Toc118057082)

[**3. Discuss how to overcome these security issues when building a secure cloud platform** 42](#_Toc118057083)

[3.1. How to Safeguard Your Data from Cloud Misconfigurations 42](#_Toc118057084)

[3.2. The Ways to Prevent Data Leaks in the Cloud 42](#_Toc118057085)

[3.3. Cloud Computing Security Services to Mitigate DDoS Attacks 43](#_Toc118057086)

[**References** 46](#_Toc118057087)

**Table of Figures**

[Figure 1: Create new repository 9](#_Toc118057508)

[Figure 2: VS code terminal statements 9](#_Toc118057509)

[Figure 3: ATN toys store repository 10](#_Toc118057510)

[Figure 4: Heroku registration page 11](#_Toc118057511)

[Figure 5: Heroku login page 12](#_Toc118057512)

[Figure 6: Create new app 13](#_Toc118057513)

[Figure 7: Accsess aplication interface 13](#_Toc118057514)

[Figure 8: Installing Heroku PostgreSQL 13](#_Toc118057515)

[Figure 9: Heroku PostgresSQL 14](#_Toc118057516)

[Figure 10: PostgresSQL information 14](#_Toc118057517)

[Figure 11: Pgadmin Download page 15](#_Toc118057518)

[Figure 12: Pgadmin interface 15](#_Toc118057519)

[Figure 13: Register-Server interface 16](#_Toc118057520)

[Figure 14: PostgreSQL browser 16](#_Toc118057521)

[Figure 15: connect.php 17](#_Toc118057522)

[Figure 16: Deploy page 17](#_Toc118057523)

[Figure 17: Connect to repository 18](#_Toc118057524)

[Figure 18: Connect to repository 18](#_Toc118057525)

[Figure 19: Deploy branch 18](#_Toc118057526)

[Figure 20: Deploy sussessfuly 19](#_Toc118057527)

[Figure 21: Result deploy 19](#_Toc118057528)

[*Figure 22: Index page* 21](#_Toc118057529)

[Figure 23: Registration page 22](#_Toc118057530)

[Figure 24: Login page 23](#_Toc118057531)

[Figure 25: Shop page 23](#_Toc118057532)

[Figure 26: Product detail page 24](#_Toc118057533)

[Figure 27: Cart page 25](#_Toc118057534)

[Figure 28: Create order page 25](#_Toc118057535)

[Figure 29: Order history page 26](#_Toc118057536)

[Figure 30: Product manger page 26](#_Toc118057537)

[Figure 31: Add new product page 27](#_Toc118057538)

[Figure 32: Add new product 27](#_Toc118057539)

[Figure 33: Update product page 28](#_Toc118057540)

[Figure 34: Update product page 28](#_Toc118057541)

[Figure 35: Suplier manager page 29](#_Toc118057542)

[Figure 36: Add new suplier page 29](#_Toc118057543)

[Figure 37: Update Suplier paga 30](#_Toc118057544)

[Figure 38: Shop manager page 30](#_Toc118057545)

[Figure 39: Add new shop page 31](#_Toc118057546)

[Figure 40: Update Shop pgae 31](#_Toc118057547)

[Figure 41: Order manager page 32](#_Toc118057548)

[Figure 42: Update order page 33](#_Toc118057549)

[Figure 43: Shop manager page 33](#_Toc118057550)

[Figure 44: Upadte Shop 35](#_Toc118057551)

[Figure 45: User manager page 36](#_Toc118057552)

**Assignment 2**

# **I. Deploy the website.**

## **1. How to deploy (put the website) to Heroku and connect to Postgres**

To be able to deploy the website to the Heroku cloud platform and connect a PostgreSQL relational database management system, we need to go through the steps of installing and connecting the platforms together. Below I will show step-by-step how to deploy my website to the Heroku cloud platfrom.

**Step 1:** Create a new repository on my Github Github profile, then I upload the entire source code of the website to the repository I just created.

* On the GitHub Repositories page click the "New" button to create a new repository.

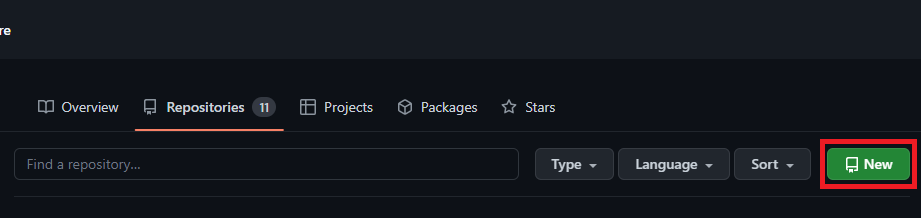


Figure 1: Create new repository

* After creating the Repository, I used the Visual Studio Code source code editor terminal to upload the source code to GitHub by executing the statements below.













Figure 2: VS code terminal statements

* Here is the output when I push successfully:

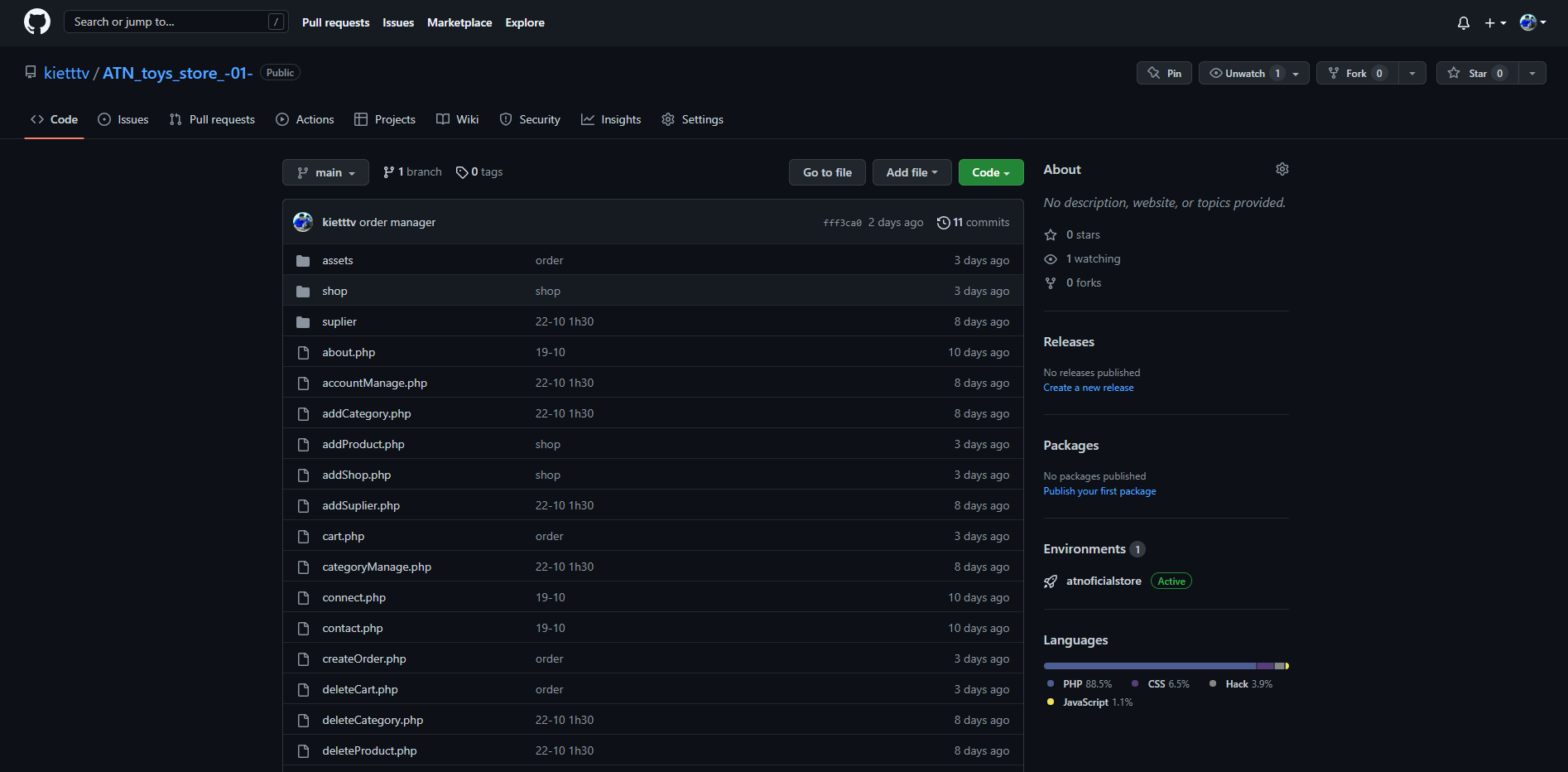


Figure 3: ATN toys store repository

**Step 2:** Register an account and log in on the Heroku cloud platform.

* Access the Heroku registration page enter all the information correctly and click the "Create Free Account" button.

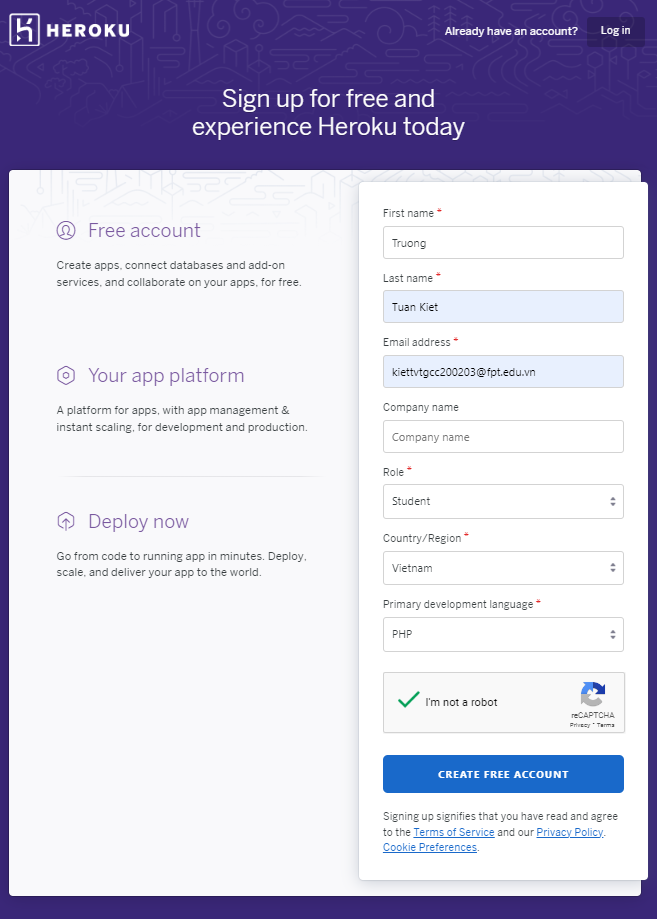


Figure 4: Heroku registration page

* After successful account registration, I access the login page and log in with the account I just registered.

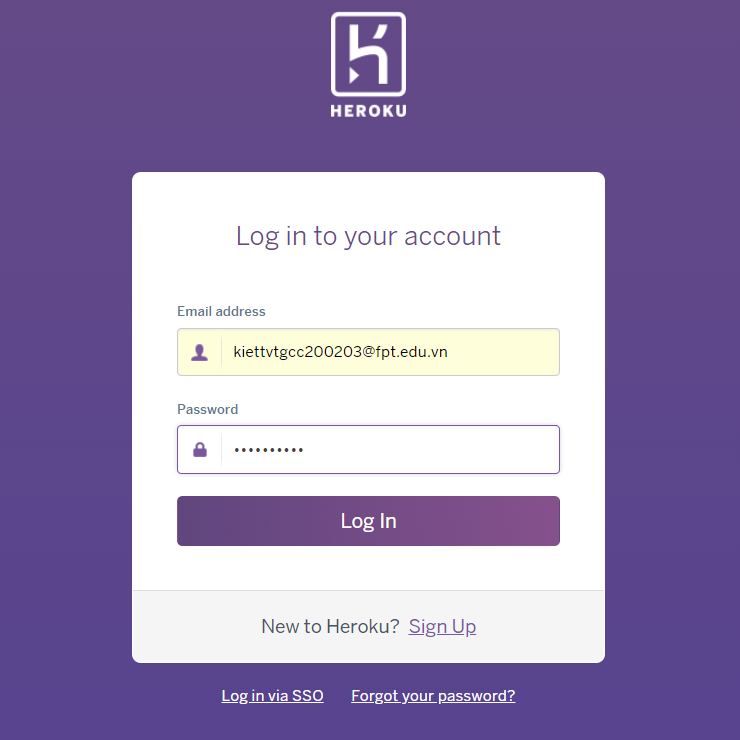


Figure 5: Heroku login page

**Step 3:** Create a new application on the Heroku cloud platform.

* After successful login, I started creating an application on Heroku by clicking the "New" button on the Heroku interface.

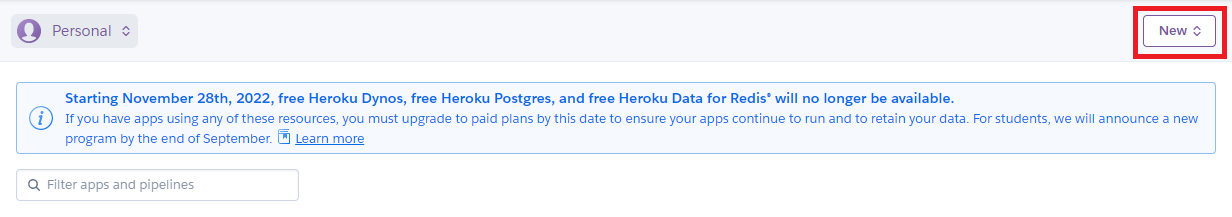


Figure 6: Create new app

* After successfully creating the application, I click on "atnoficialstore" to access the application interface.

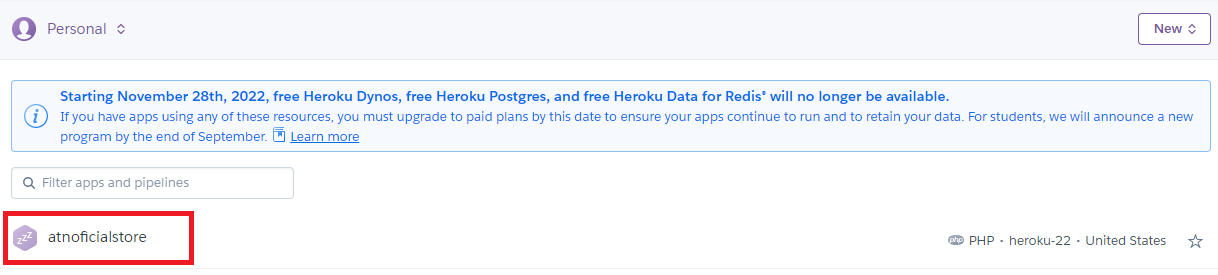
****

Figure 7: Accsess aplication interface

**Step 4:** Install Heroku Postgresql add-ons to Heroku application.

* Next, I started installing Heroku PostgresSQL, I clicked select Resource then click "Find more add-ons" button to select database. I will choose Heroku PostgresSQL.

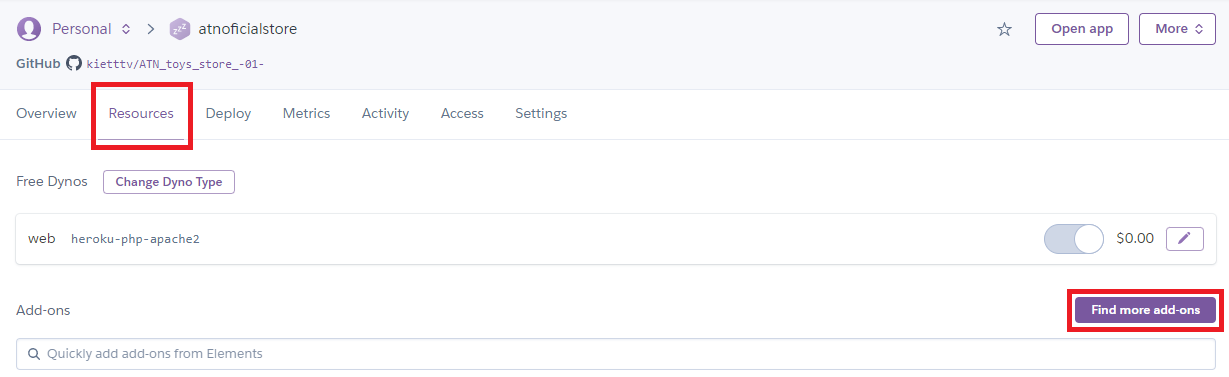
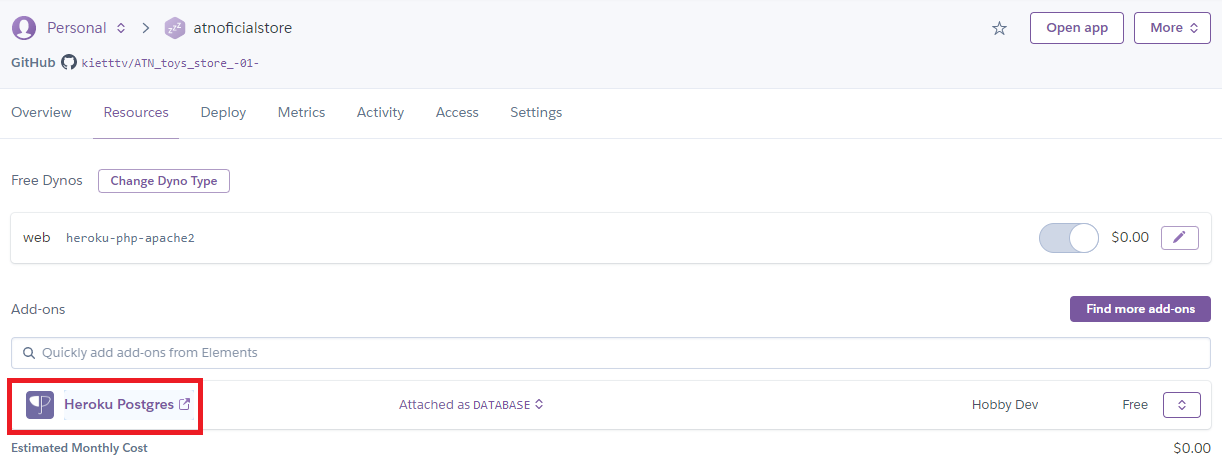


Figure 8: Installing Heroku PostgreSQL

* After installing Heroku PostgresQL successfully, I access the settings of PostgresQL to get all the information about the database by clicking on the hyperlink PostgresQL on the app interface.



* Click the "Settings" option then clicks the "View Credentials..." button to view PostgresSQL information.

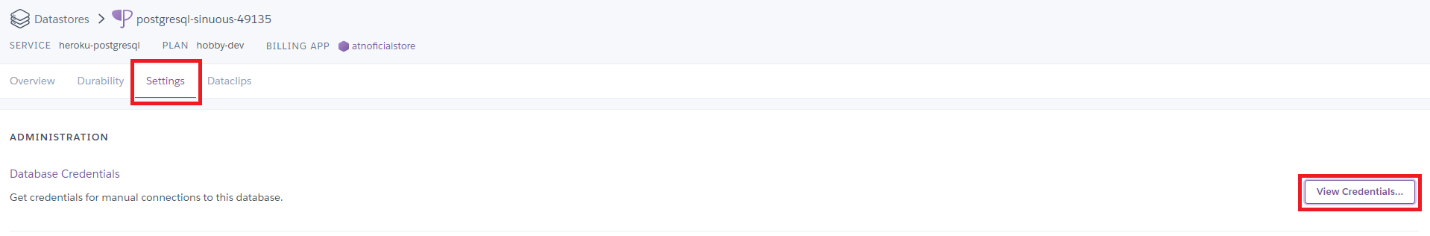


Figure 9: Heroku PostgresSQL

* Here PostgresSQL infomation

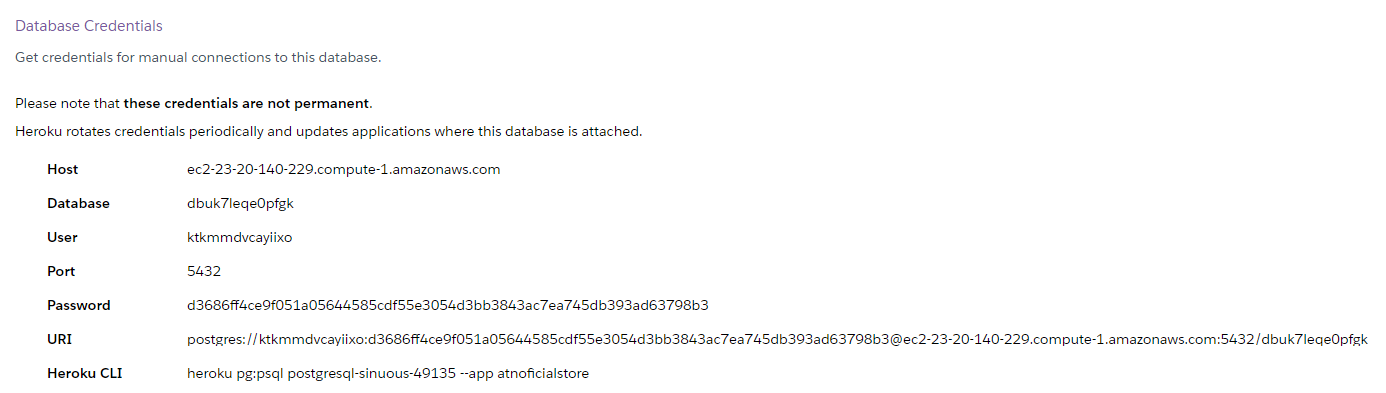


Figure 10: PostgresSQL information

**Step 5:** Install and configured Pgadmin.

* Access the Pgadmin Download page to download and install the Pgadmin PostgreSQL Database management tool.

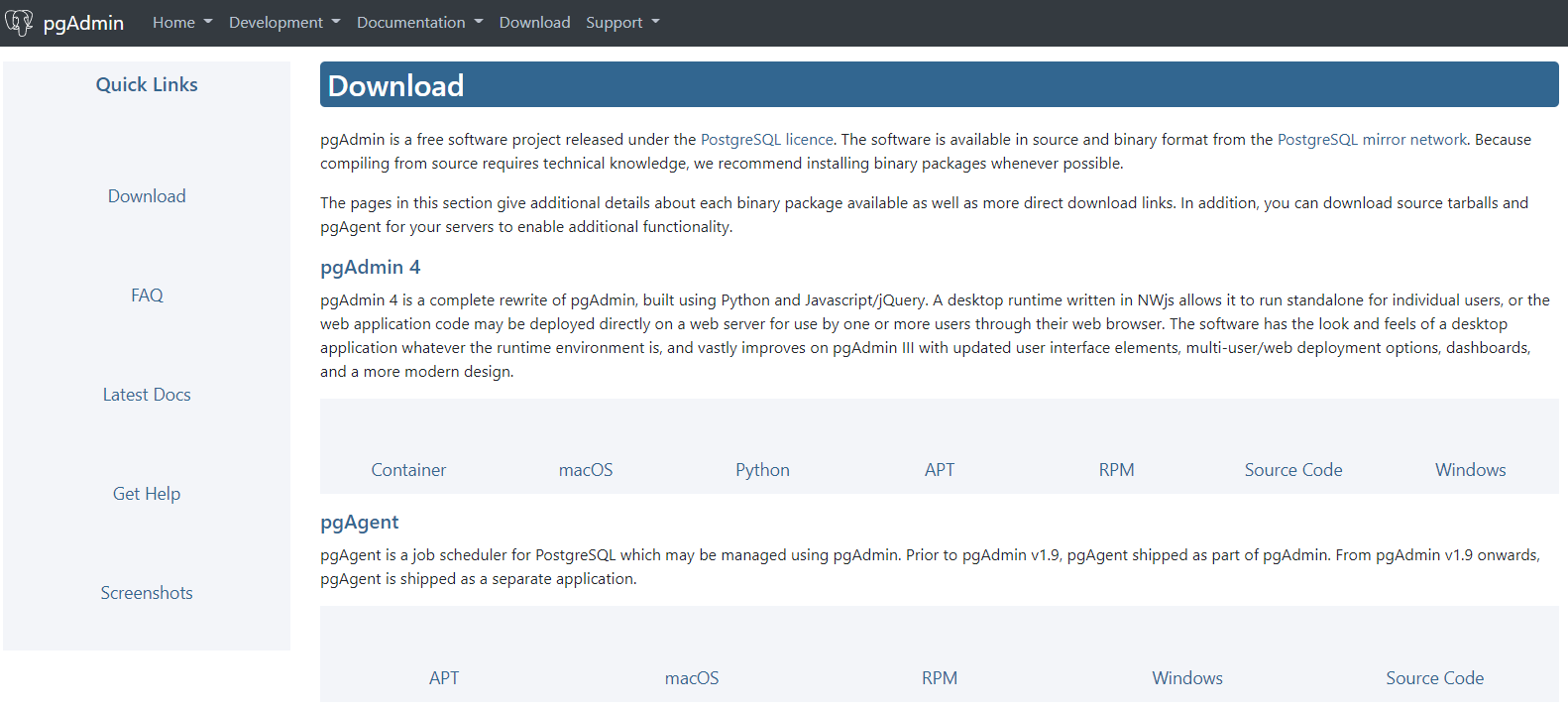


Figure 11: Pgadmin Download page

* After successfully installing Pgadmin, I started creating a server for the PostgreSQL database by clicking the Dashboard option on the navigation bar and selecting the "Add New Server" icon.

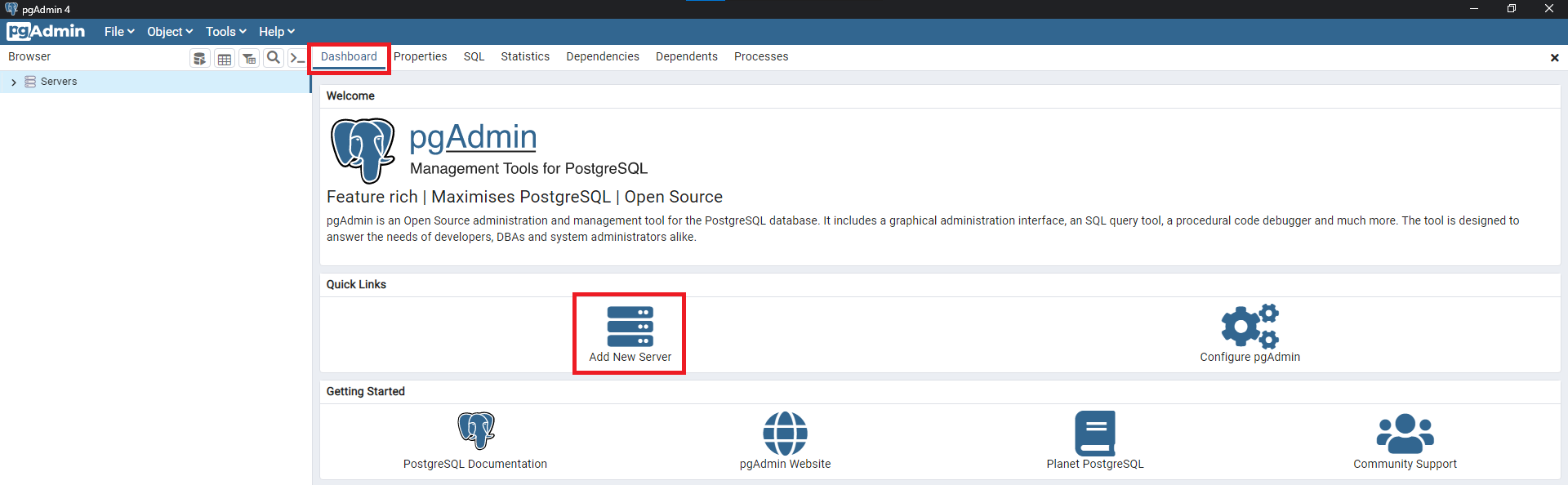


Figure 12: Pgadmin interface

* Entering all the information Heroku provided about the database and then click “Save” button.

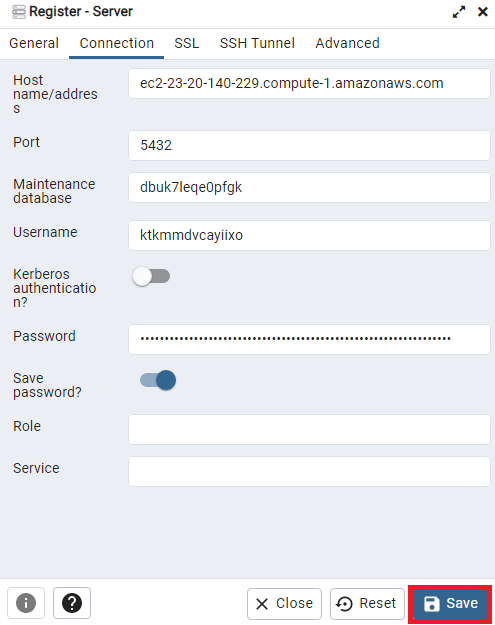


Figure 13: Register-Server interface

* After successfully creating the server, I scrolled in the browser column to find my database. Here we can create, edit, and delete tables; create, edit, and delete columns of each table.

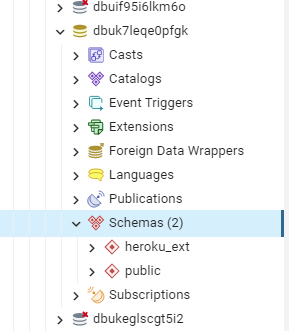


Figure 14: PostgreSQL browser

* Connection PostgreSQL database in my source code project. I create file connect.php declare variables "host", "dbname", "user", "password", "port", to store the information received from Heroku. Then I declared the variable "conn" with the function "pg\_connect()", to connect to PostgreSQL I passed the variables declared above into the function "pg\_connect()".

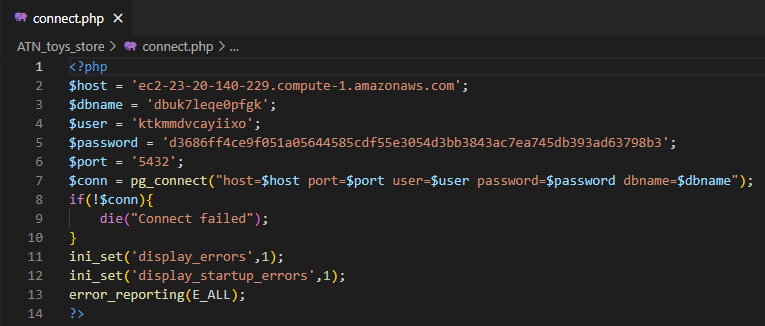


Figure 15: connect.php

**Step 6:** Connect GitHub and deploy the website.

* Click the Deploy option on the navigation bar to go to the Deploy page and click the "Connect to Github" button the system will use the browser cookie and connect to my GitHub account.

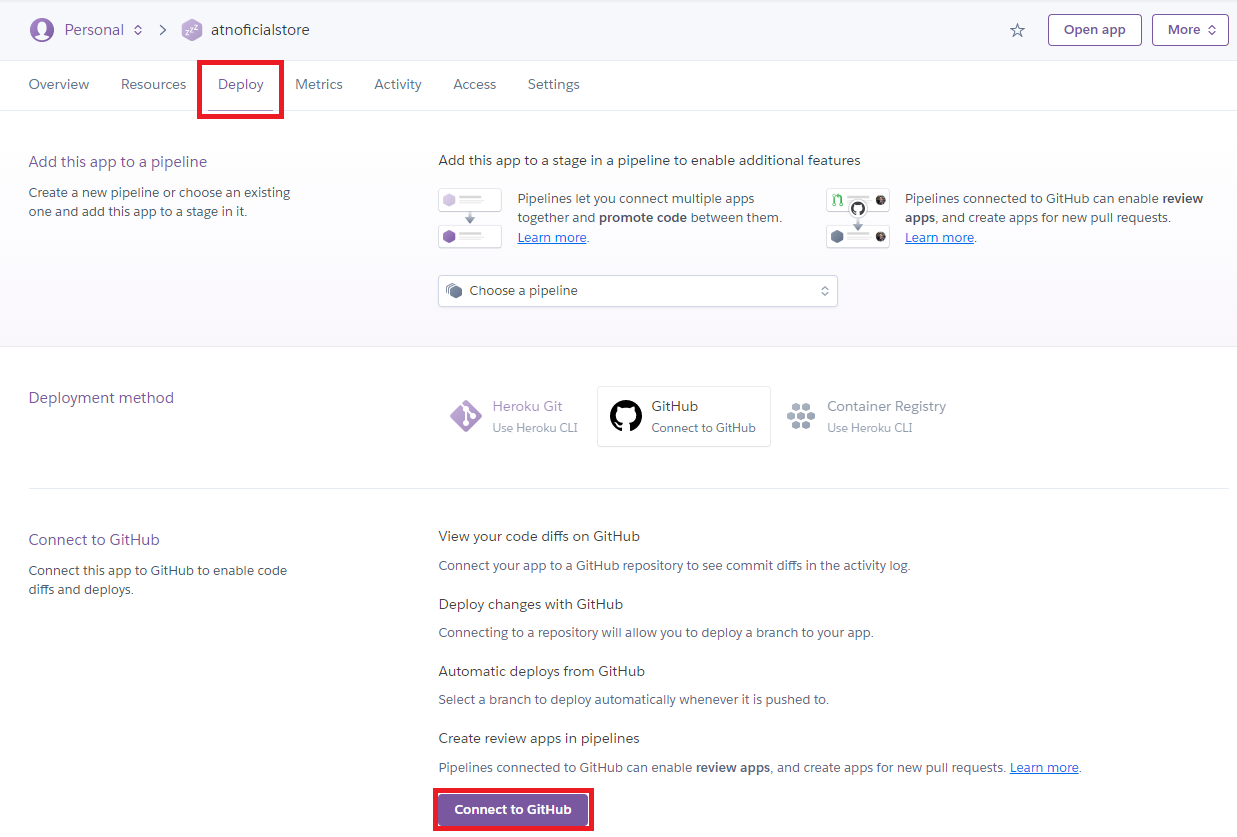


Figure 16: Deploy page

* After connecting to my GitHub account successfully, I continue to connect to the repository where I pushed my source code by entering the repository name in the Search bar and clicking the Search button.

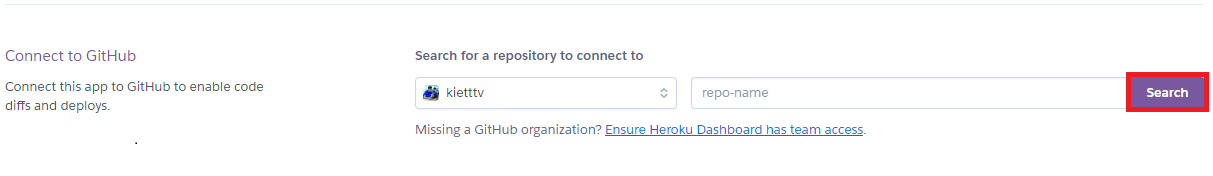


Figure 17: Connect to repository

* After successfully searching the repository, I click the "Connect" button to connect to the repository.

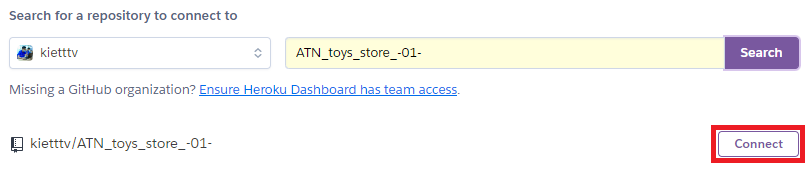


Figure 18: Connect to repository

* Next, I start deploying the website after I was successfully connected. By clicking the "Enable Automatic Deploys" button and the "Deploy Branch" button.

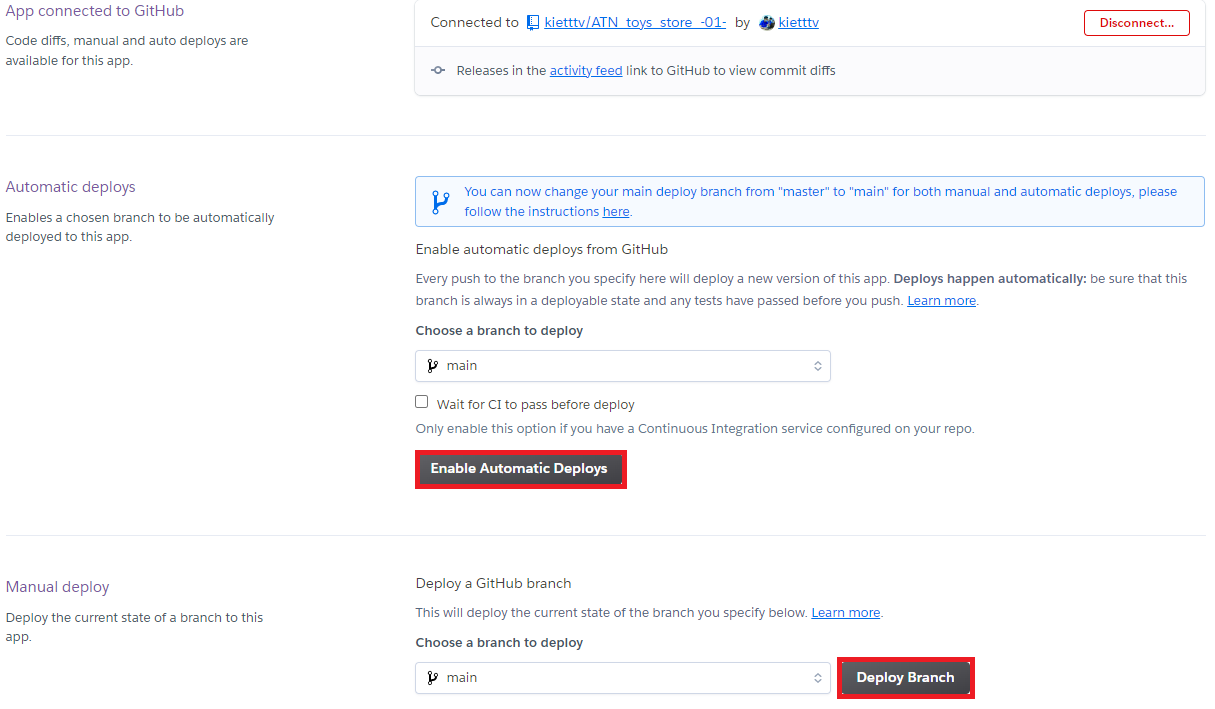


Figure 19: Deploy branch

* After successful deployment, I can click the "View" button to see the result.

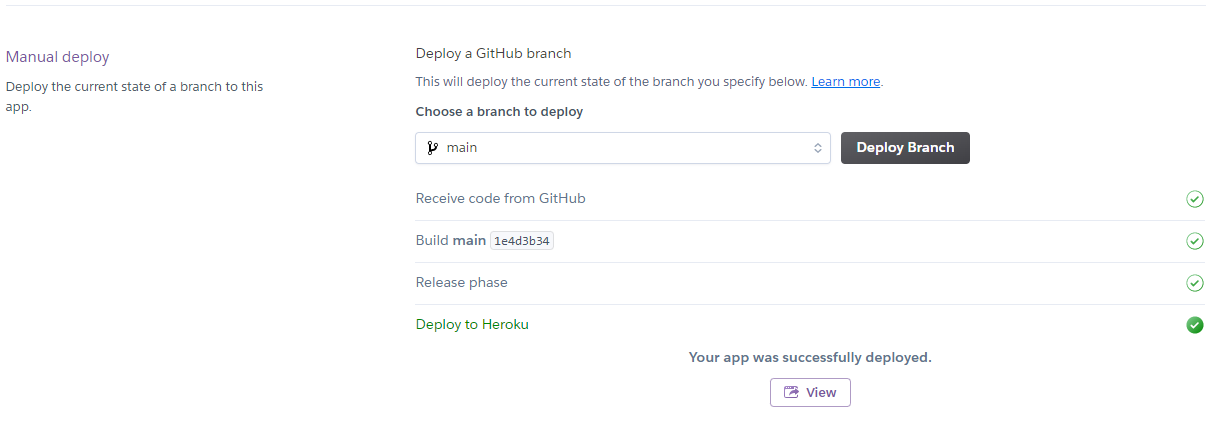
****

Figure 20: Deploy sussessfuly

* Here result.

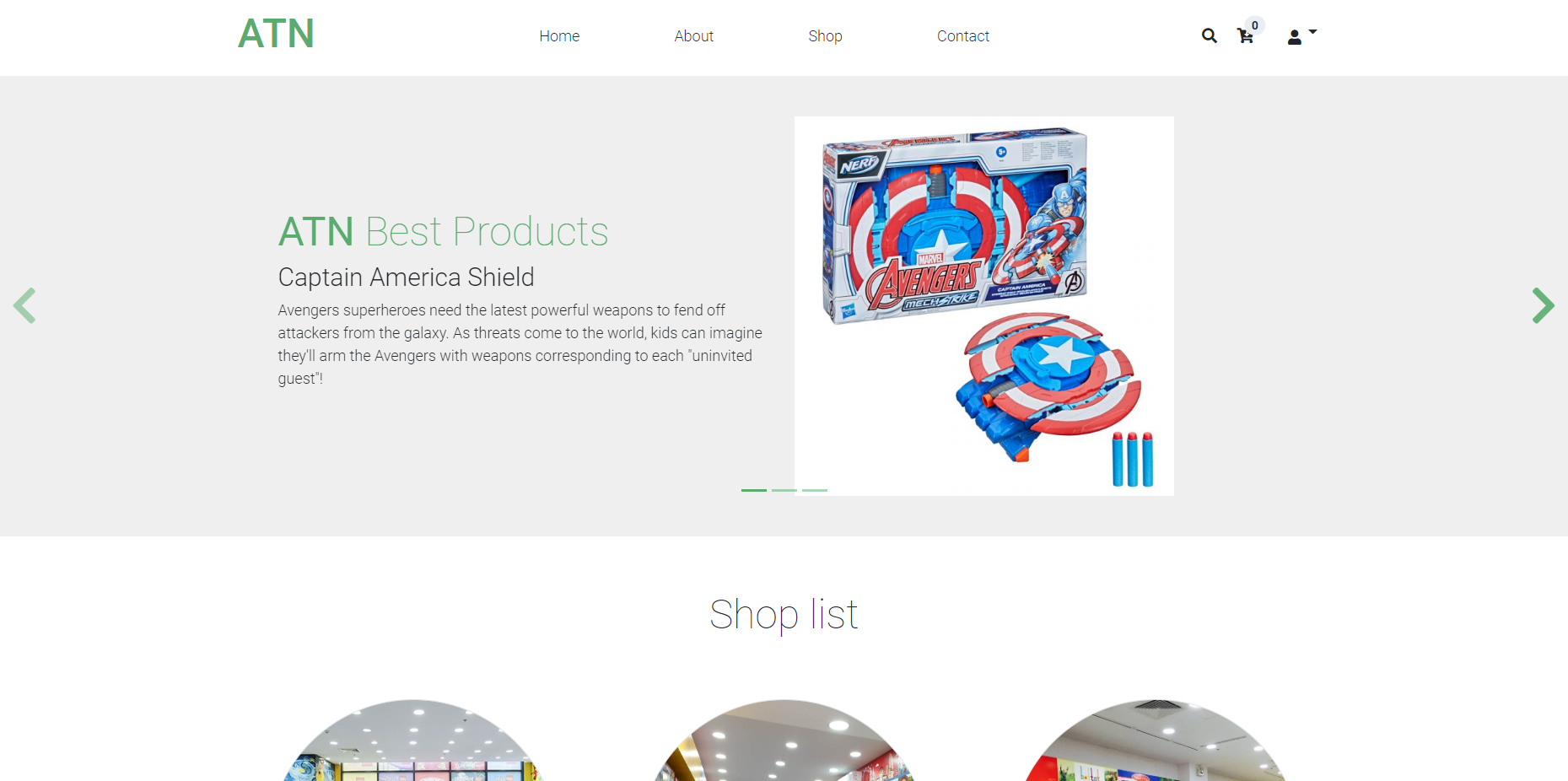


Figure 21: Result deploy

## **2. Provide pictures of implemented functions**

### **2.1. Over view**

Link GitHub: <https://github.com/kietttv/ATN_toys_store_-01-.git>

Link Website: <https://atnoficialstore.herokuapp.com/index.php>

Acccount:

|  |  |  |
| --- | --- | --- |
| **Role** | **User name** | **Password** |
| Admin | admin | admin |
| User | user | user |

Web Application has the several function that are:

User:

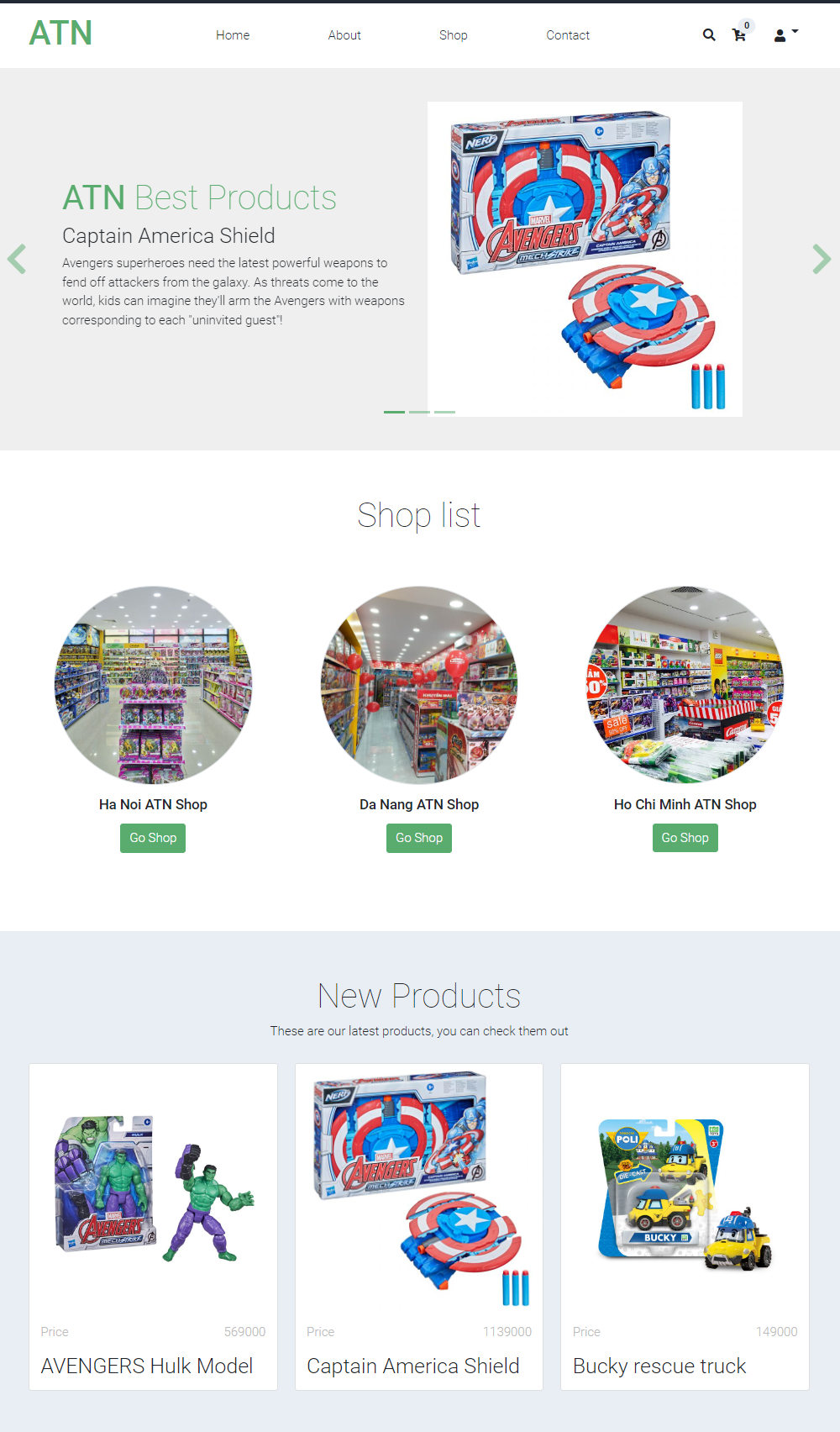
* Register new account,
* Login,
* Logout,
* Search product by name,
* View list products,
* View product detail,
* Add product into cart,
* View cart,
* Craete order,
* Payment.

Admin:

* Includes user functions,
* Manegement product (add new product, update product infomation, delete product),
* Manegement suplier (add new suplier, update suplier infomation, delete suplier),
* Manegement shop (add new shop, update shop infomation, delete shop),
* Manegement Order (update Order infomation, confirm order, delete order),
* Manegement shop (add new shop, update shop infomation, delete shop),
* Manegemrnt account (View user account infomation).

### **2.2. Index**

The index page is the first interface that appears when customers visit the website. It will display a list of the best products, pictures of the system's stores, and a list of the latest products. Customers can access and view the best products and latest products in detail.

****

*Figure 22: Index page*

### **2.3. Registration page**

To be able to add products to the cart and place an order, customers need to create an account and log in to the website. This is a registration page where customers can provide customer information and create an account to be able to log in.

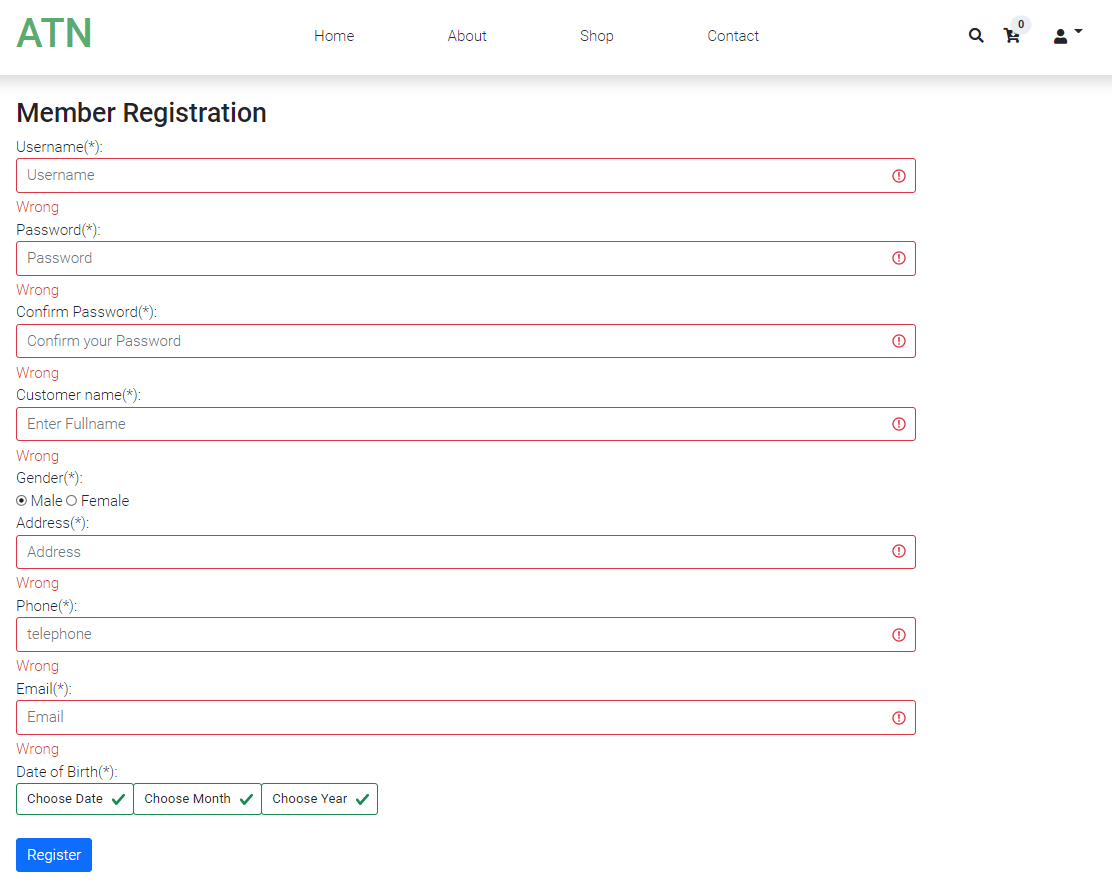
****

Figure 23: Registration page

### **2.4. Login**

After successfully creating an account, the customer can log in to the website at the login page and start adding products to the cart and placing an order.

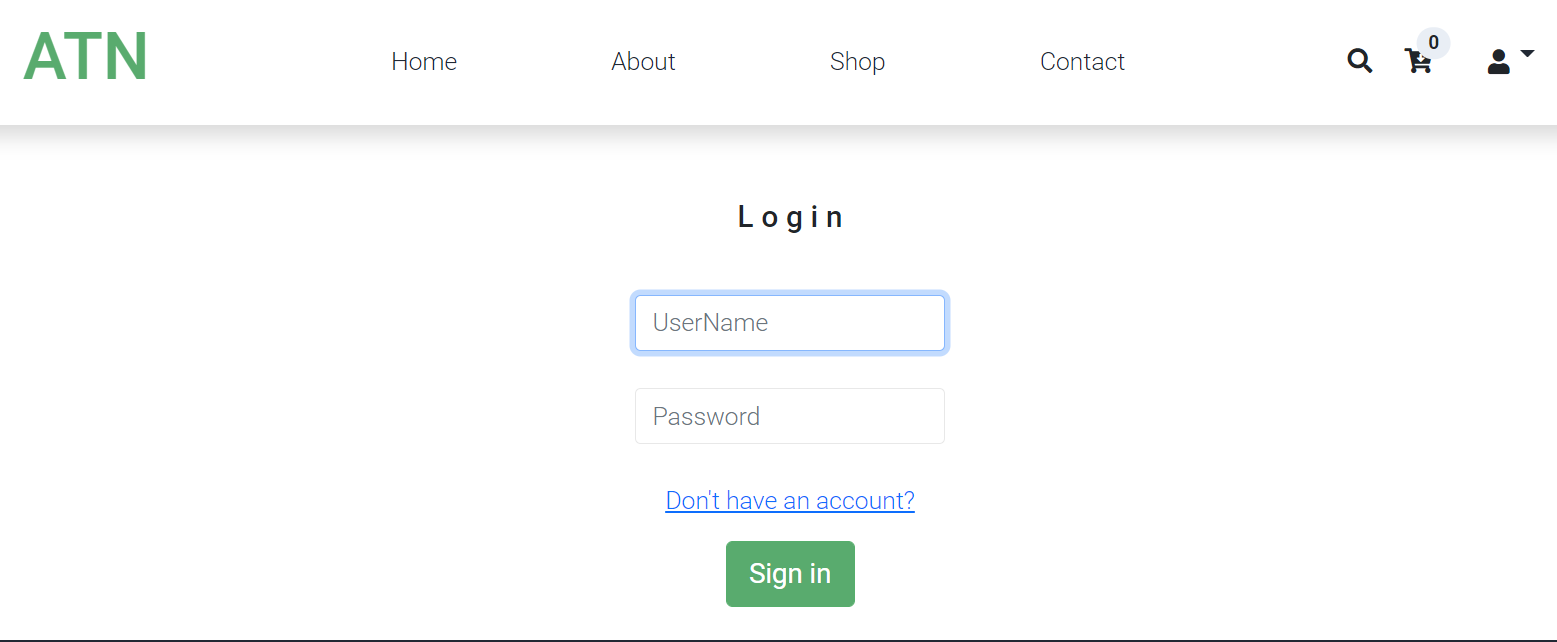


Figure 24: Login page

### **2.4. List of products**

Shop page where all the products of the system are displayed, where customers can view all products. Customers can click on each product to see detailed information about the products.

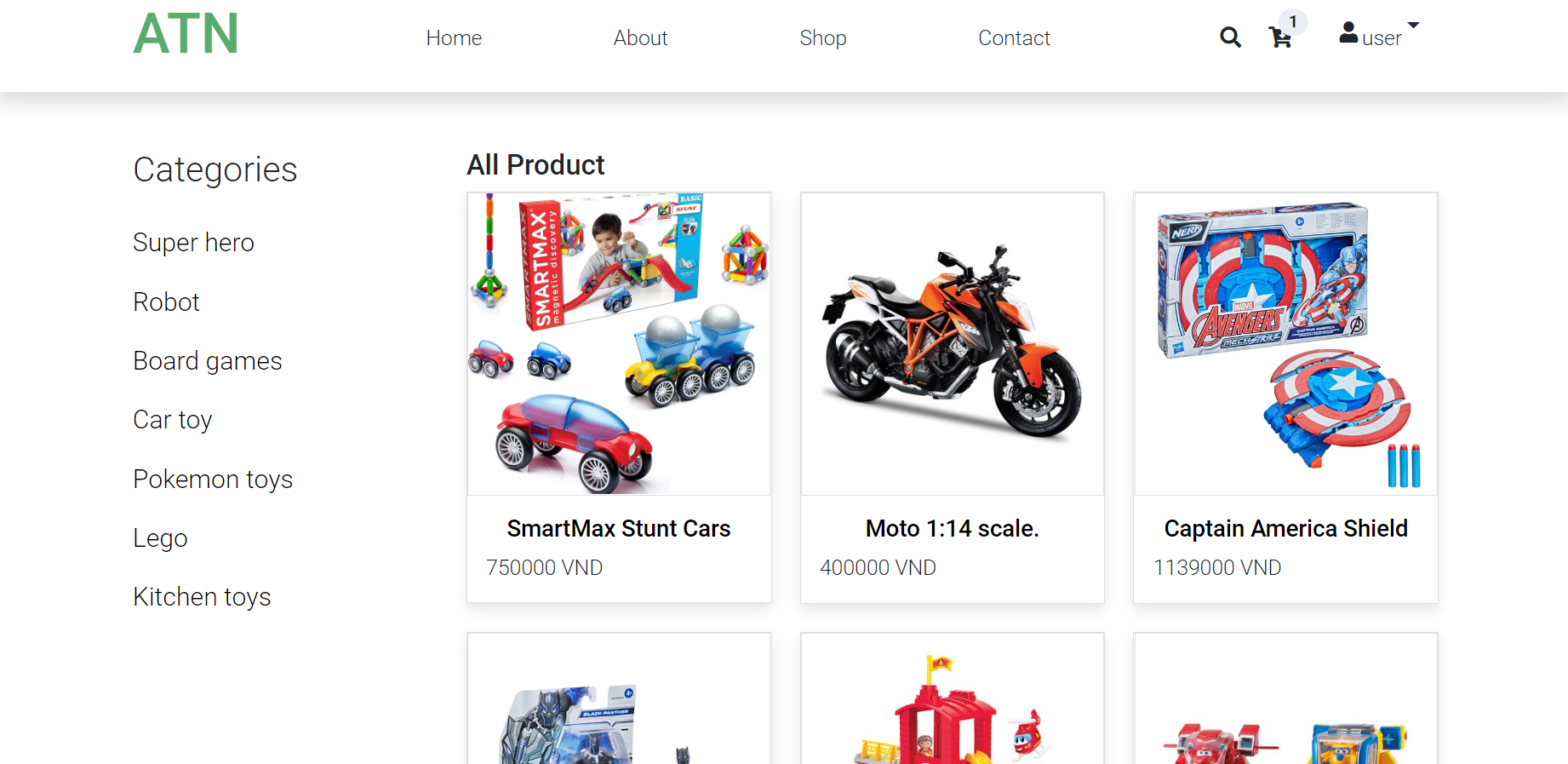
****

Figure 25: Shop page

### **2.6. Product detail**

After accessing the Shop page, the customer can click on each product to go to the details page of the product the user has selected. In this page user can select quantity and click "Add to cart" button to add product to cart.

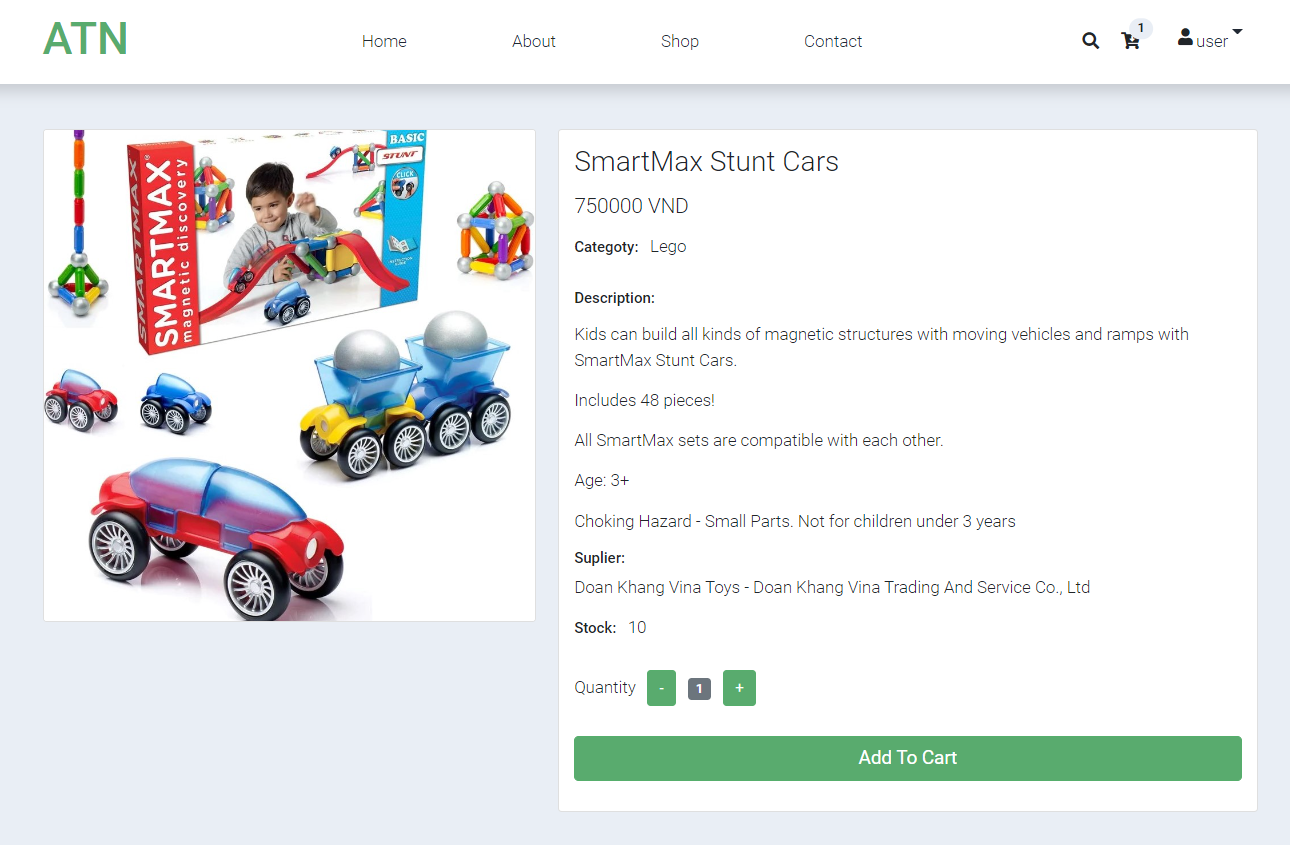


Figure 26: Product detail page

### **2.7. Cart page**

After selecting the number of products and clicking the "Add to cart" button, customers can access the cart page to view the products information that customer has added to the cart, delete products from cart, and click “Payment” to create order.

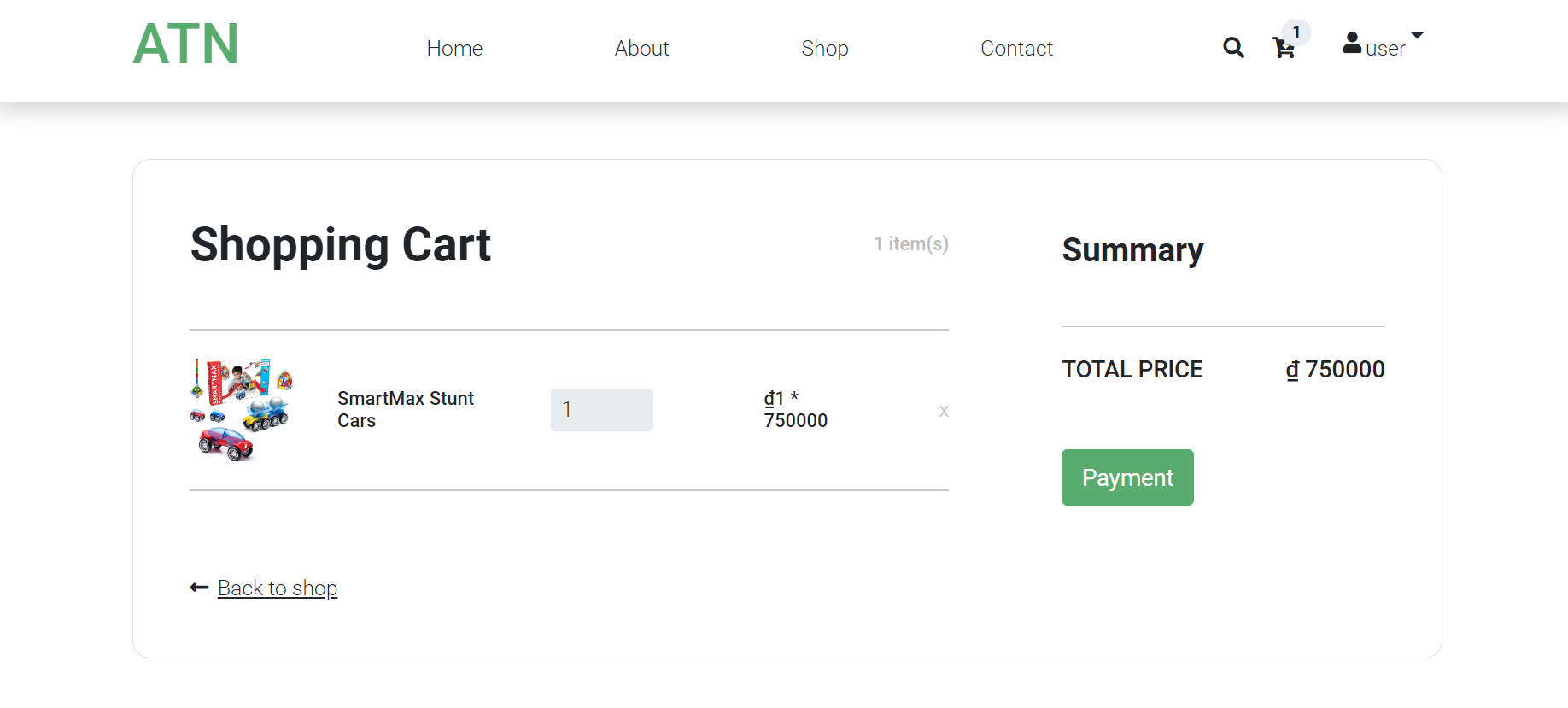
****

Figure 27: Cart page

### **2.8. Create order**

After the user clicks the "Checkout" button, the user goes to the Create Order page. At Create Order page, customer can edit order information and click "Place Order" button to place an order.

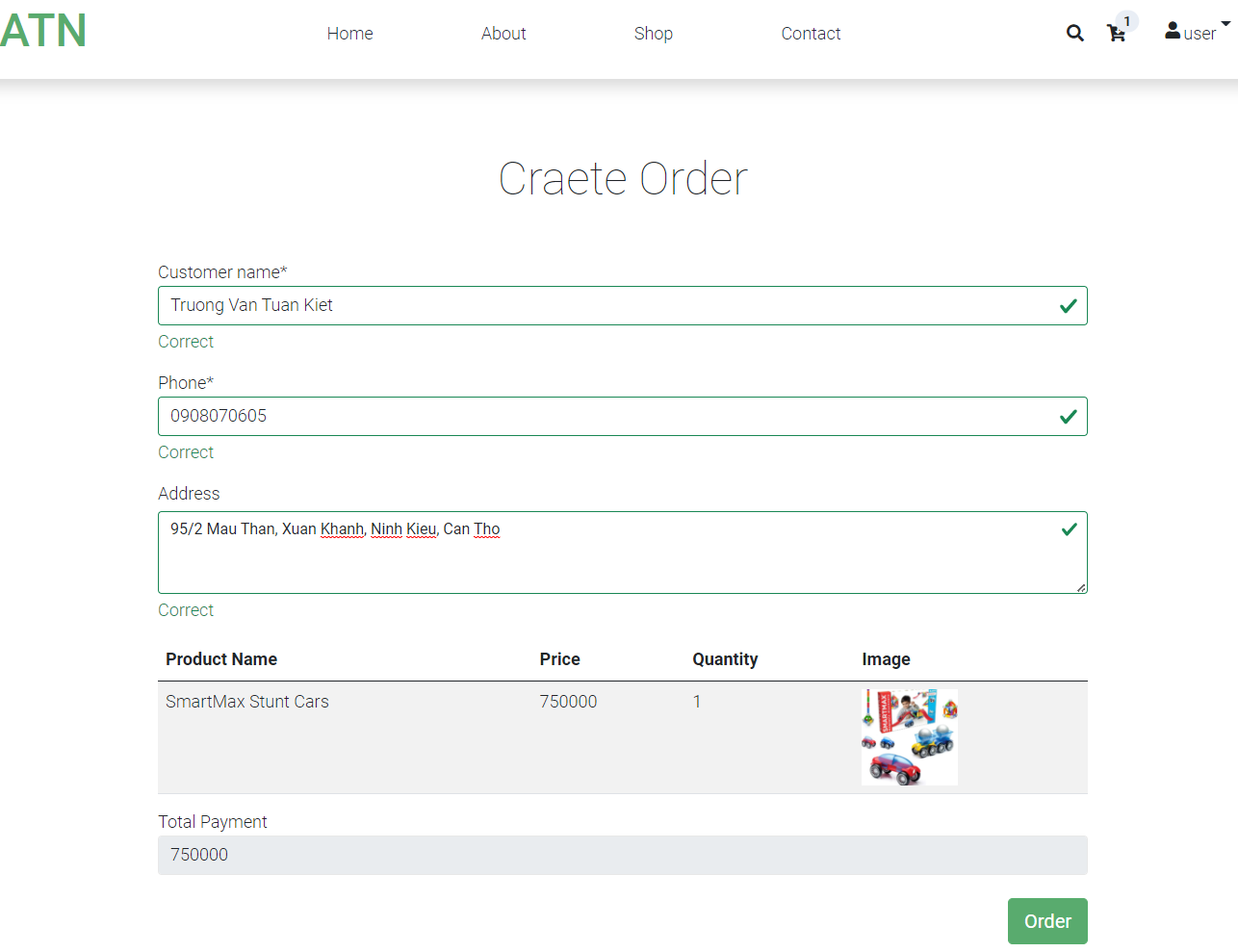
****

Figure 28: Create order page

### **2.9. Order history**

After ordering successfully, the customer can access the Order history page to see the orders that the customer has placed.

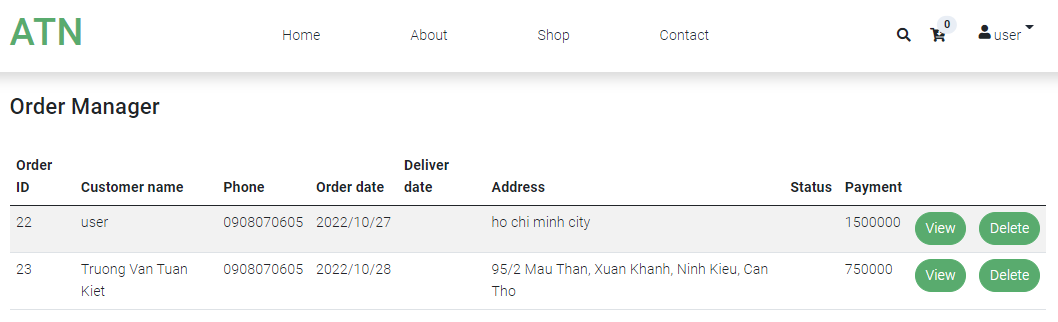


Figure 29: Order history page

### **2.10. Manegement product**

* **Manegement product page**

Product Manager page where all the products of the system are displayed. Admin can view, add new products, update product information and delete products.

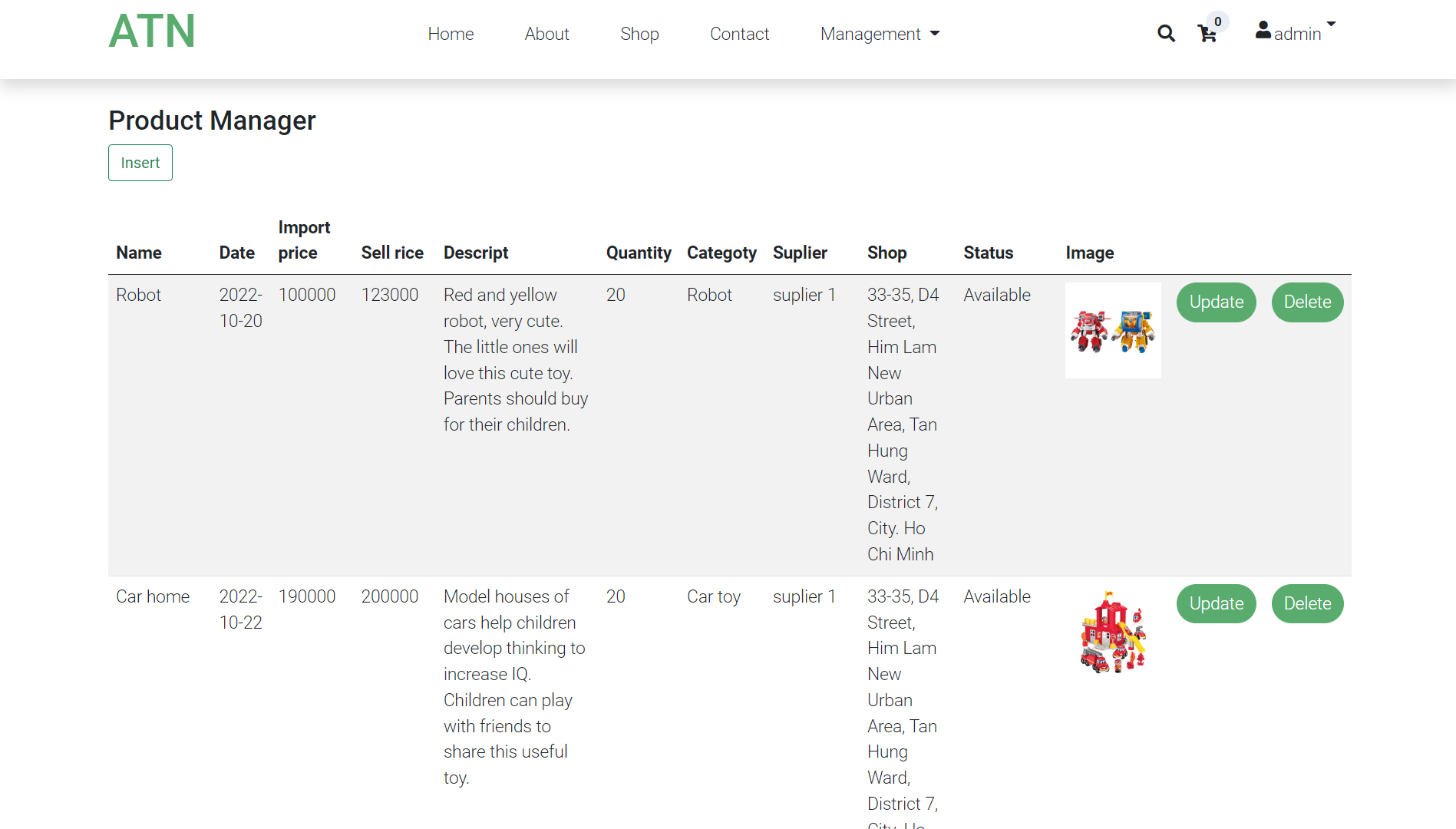
****

Figure 30: Product manger page

* **Add new product**

After clicking the insert button on the Product Manager page, the admin will access the Add new product page. On this page, the admin can enter product information and click the "Add product" button to add products to the system.

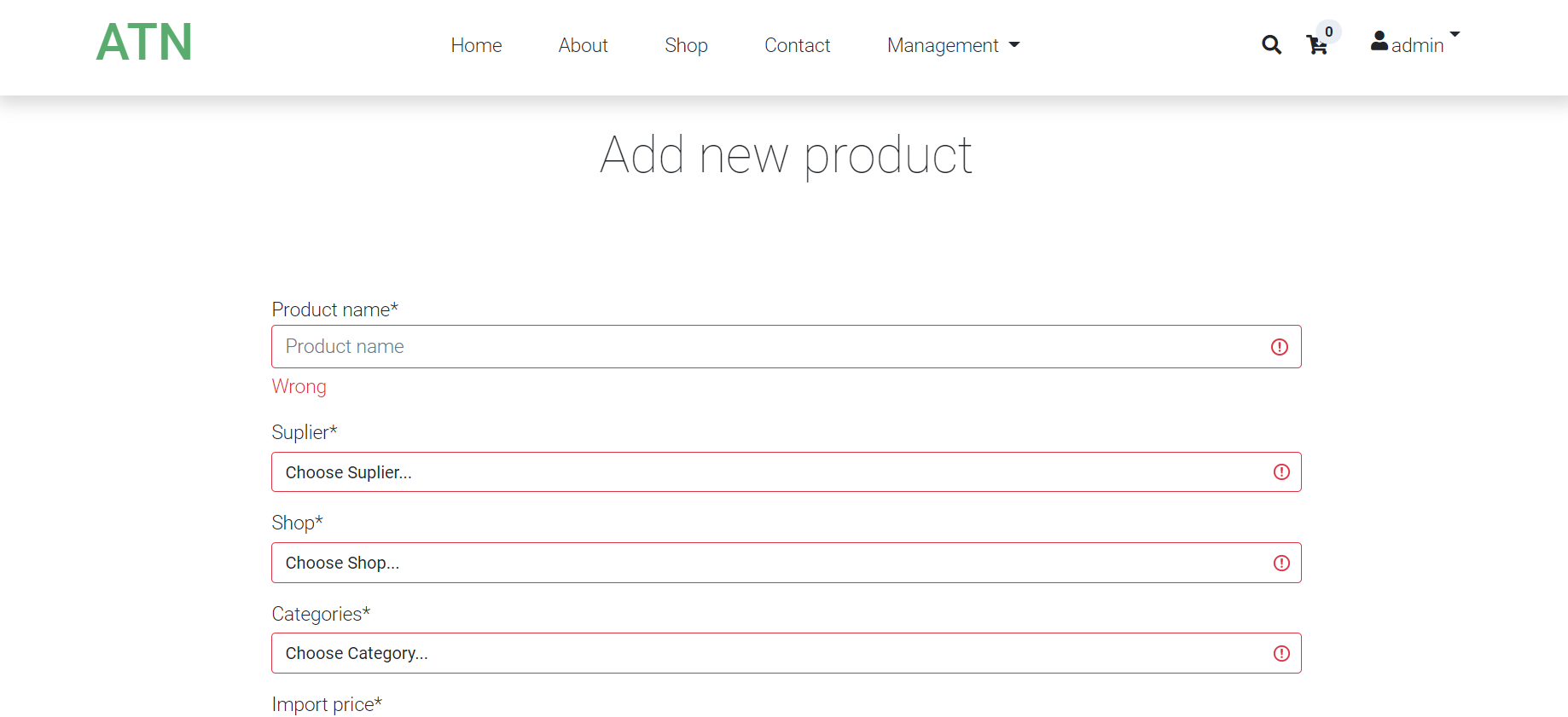


Figure 31: Add new product page

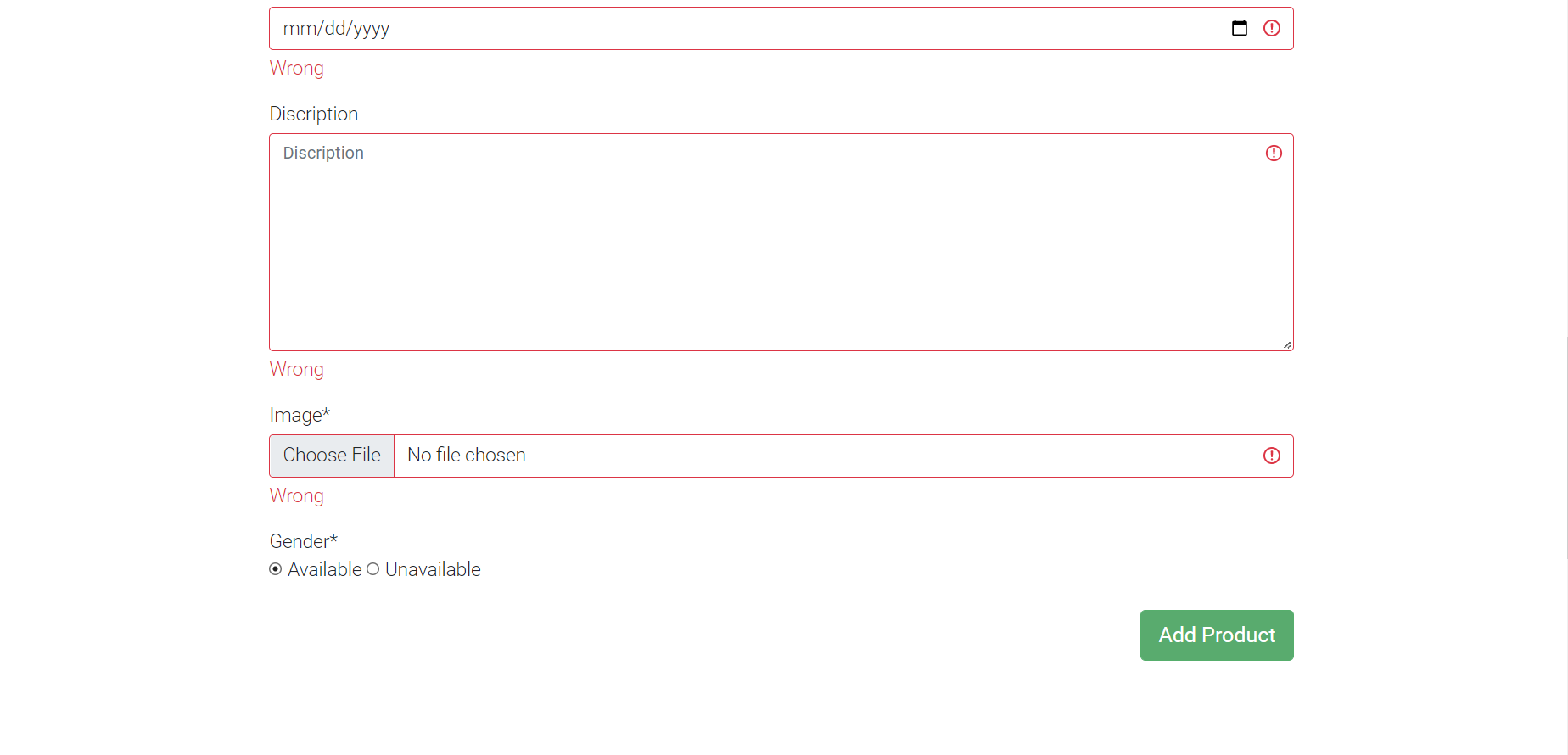


Figure 32: Add new product

* **Update product page**

After clicking the "Update" button on the Product Management Admin page will access the product update page. On the Product Update page, the admin can change product information and click the "Update Product" button for product update information.

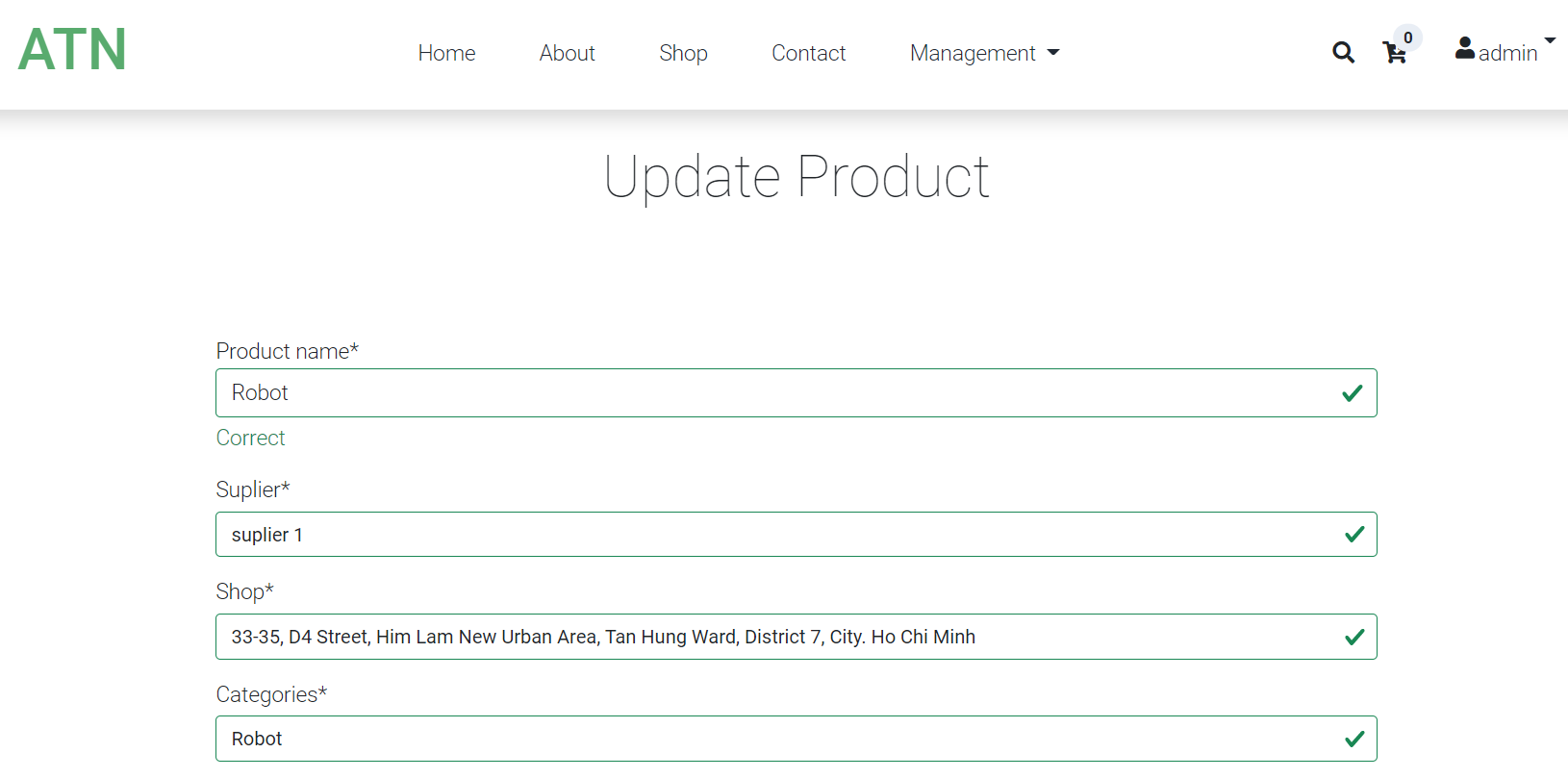


Figure 33: Update product page

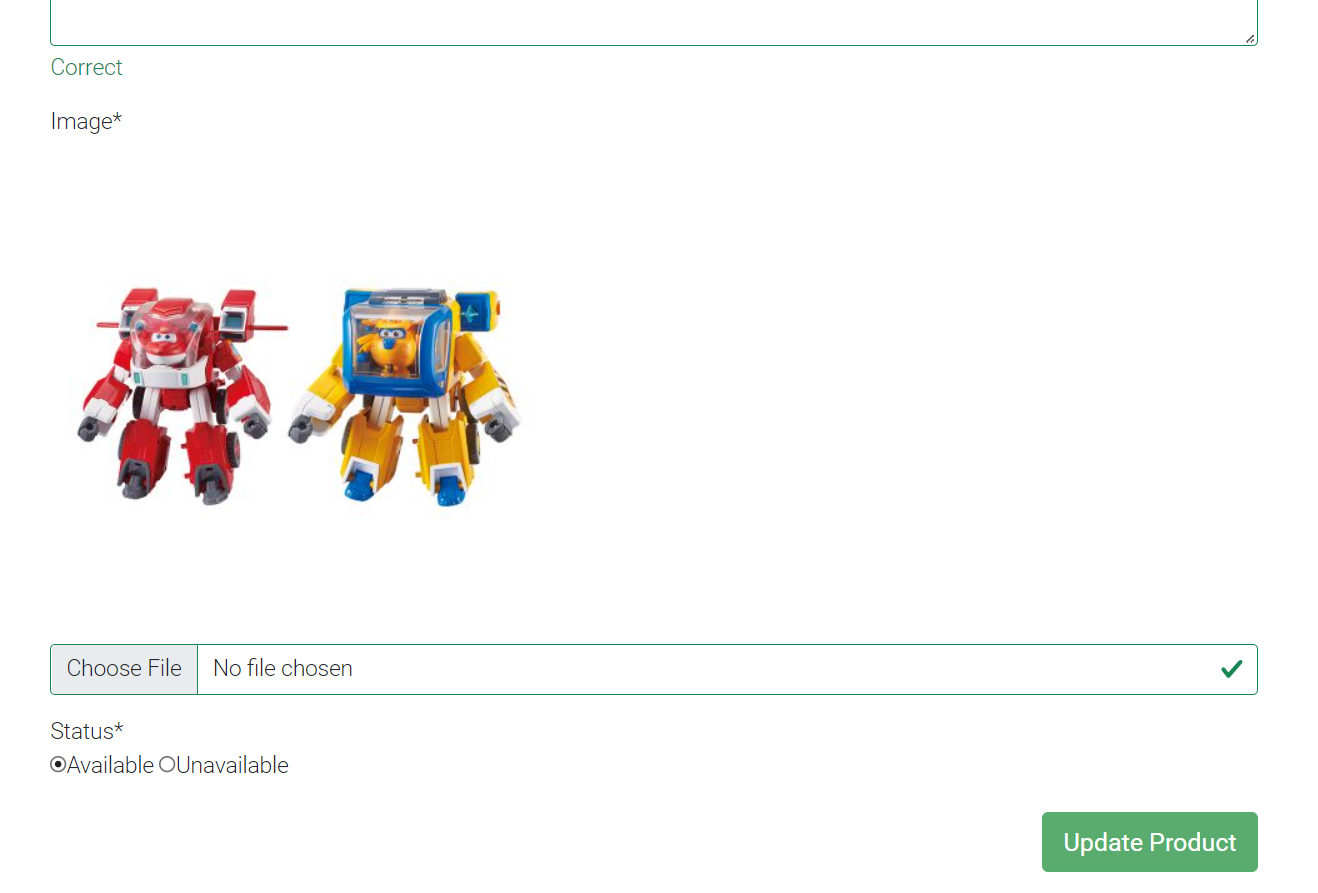
**

Figure 34: Update product page

### **2.11. Manegement suplier.**

* **Manegement suplier page**

The Suplier Manager page where all the system's supliers are displayed. Admin can view, add new suplier, update suplier information and delete suplier.

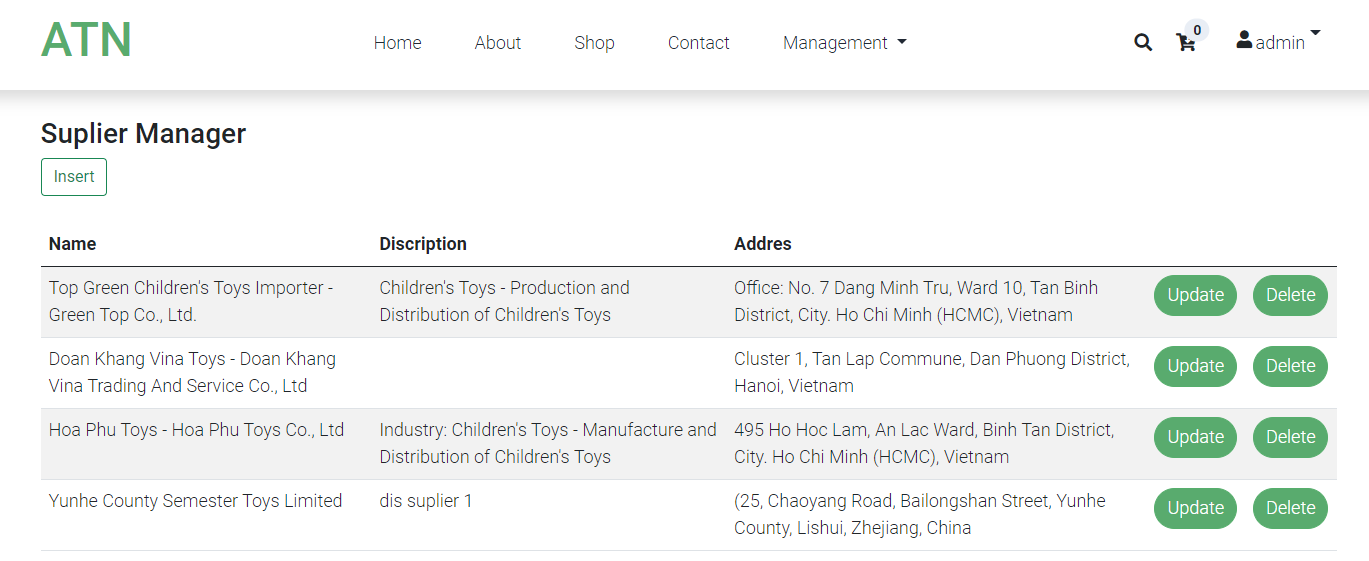
****

Figure 35: Suplier manager page

* **Add new suplier**

After clicking the insert button on the Suplier Manager page, the admin will go to the Add New Suplier page. On this page, admin can enter suplier information and press the button “Add Suplier” to add Suplier to the system.

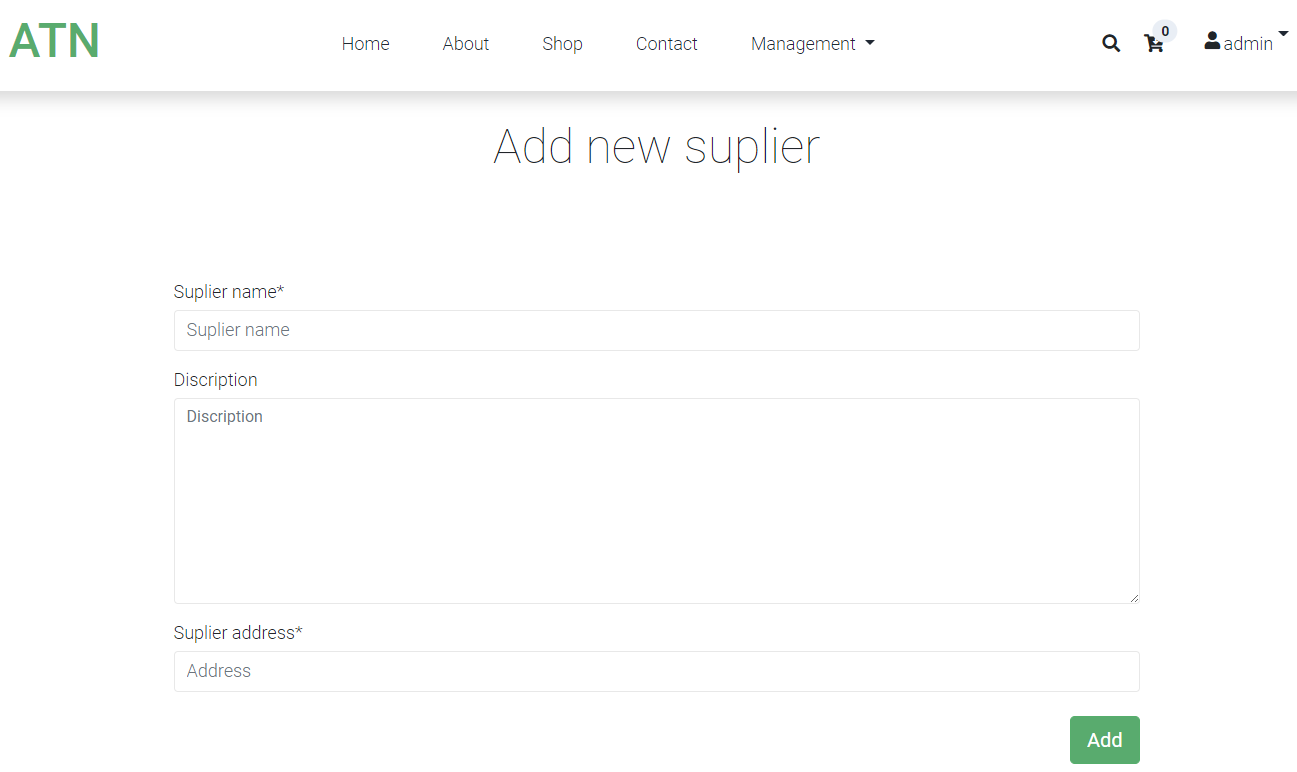
****

Figure 36: Add new suplier page

* **Update suplier**

After clicking the insert button on the Suplier Manager page, the admin will go to the Add New Suplier page. On this page, admin can enter suplier information and press the button “Add Suplier” to add Suplier to the system.

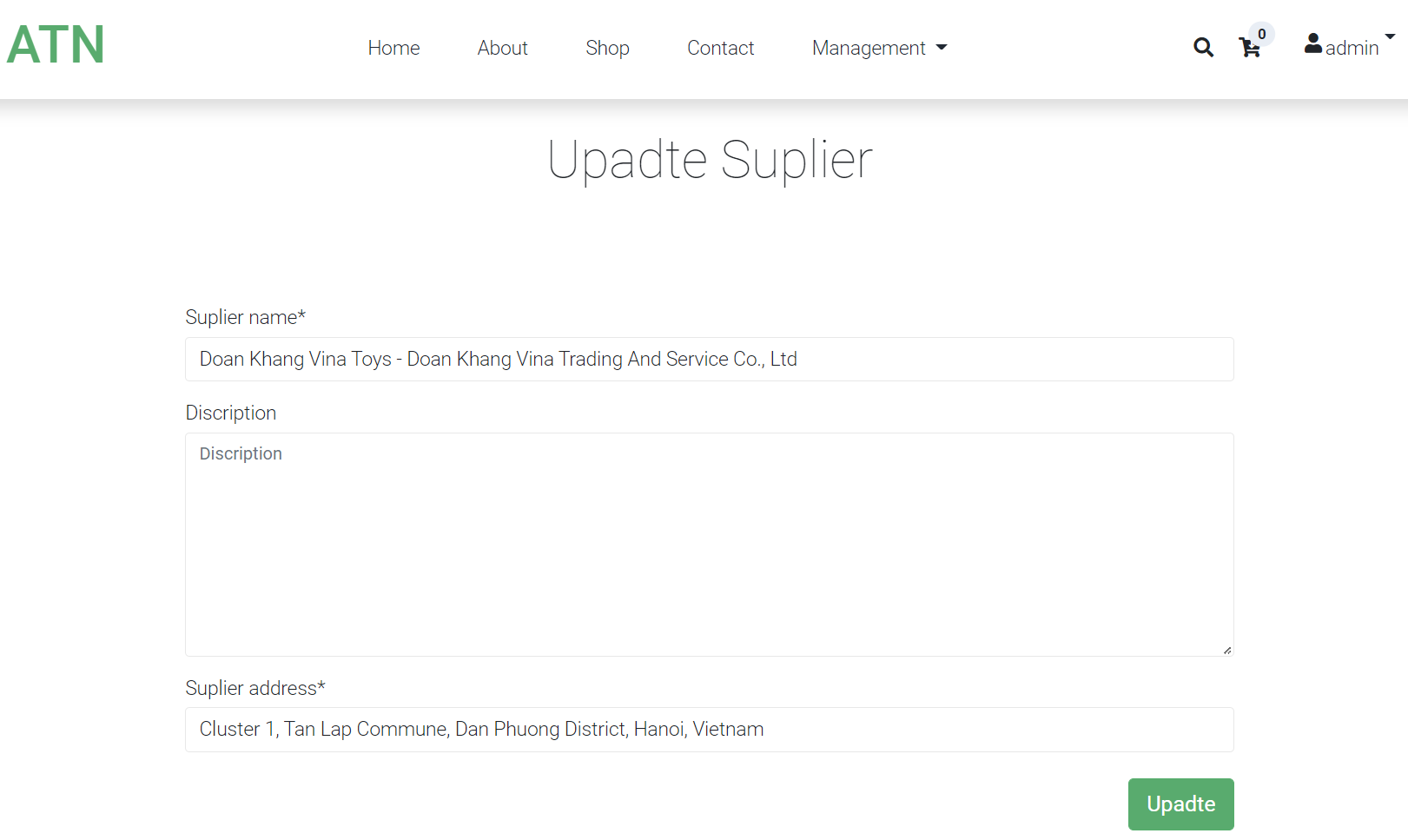
****

Figure 37: Update Suplier paga

### **2.12. Manegement shop.**

* **Manegement shop page**

Shop Manager page where all the Shops of the system are displayed. Admin can view, add new shop, update shop information and delete shop.

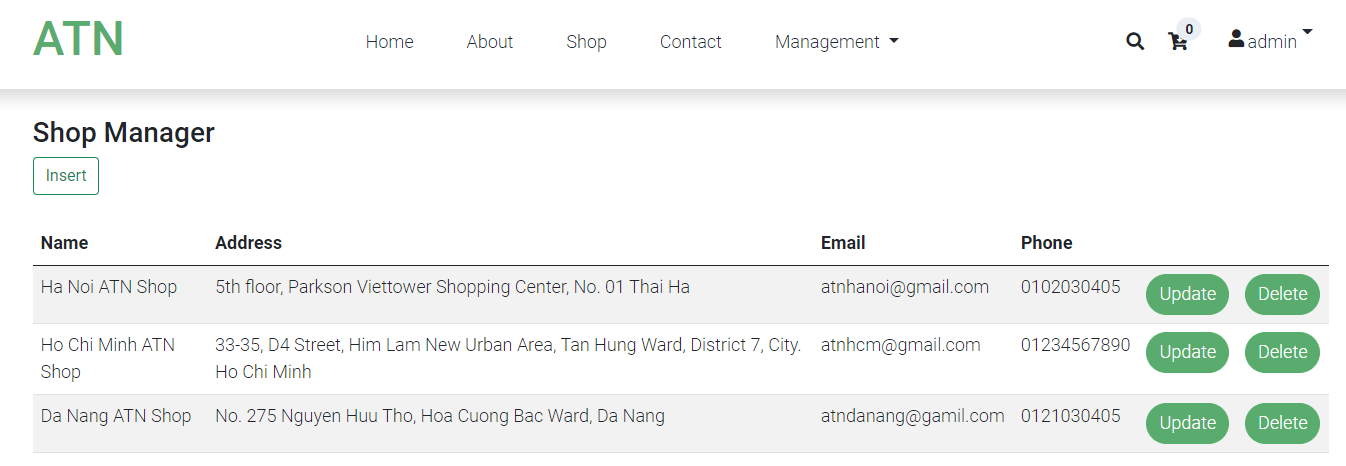
****

Figure 38: Shop manager page

* **Add new shop**

After clicking the insert button on the Shop Manager page, the admin will go to the Add New Shop page. On this page, admin can enter shop information and click "Add Shop" button to add Shop to the system.

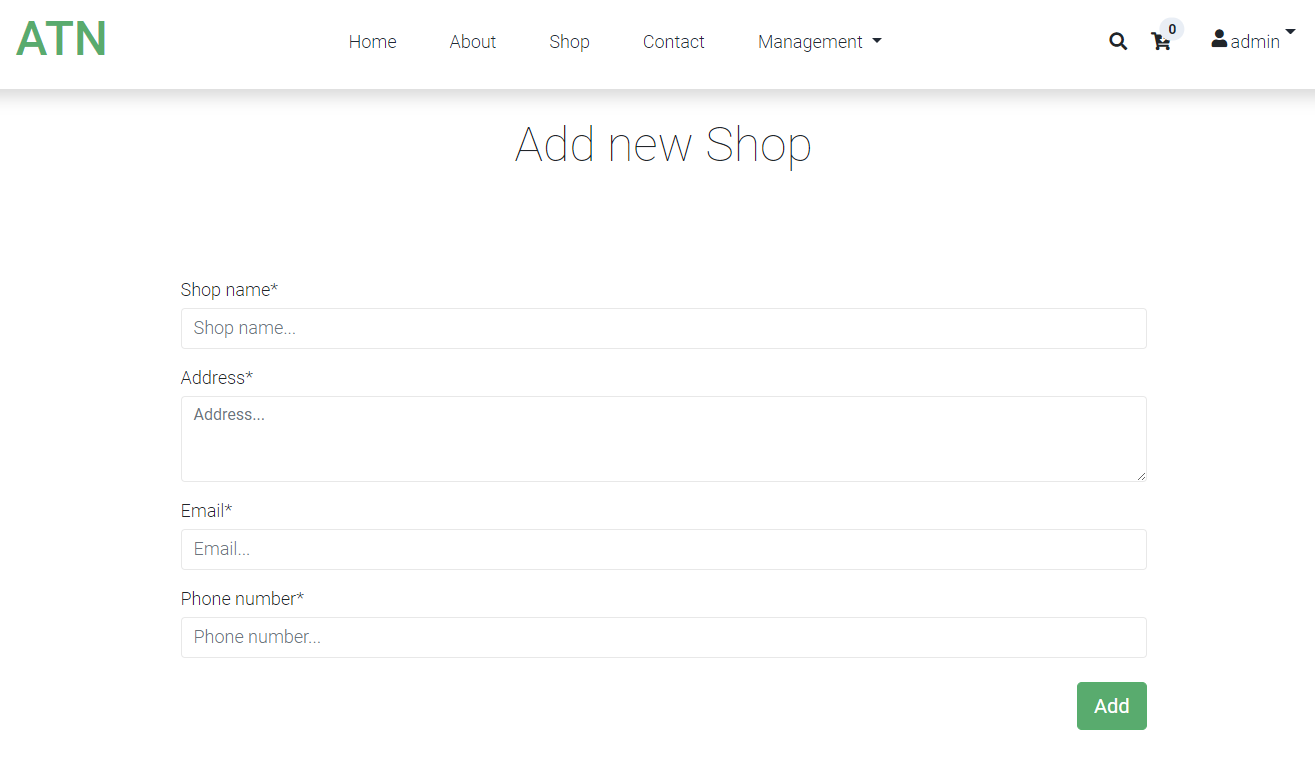
****

Figure 39: Add new shop page

* **Update shop**

After clicking the "Update" button on the Shop Manager page Admin will access the Update Shop page. At the Update Shop page, the admin can change the shop information and click the "Update Shop" button to update the shop information.

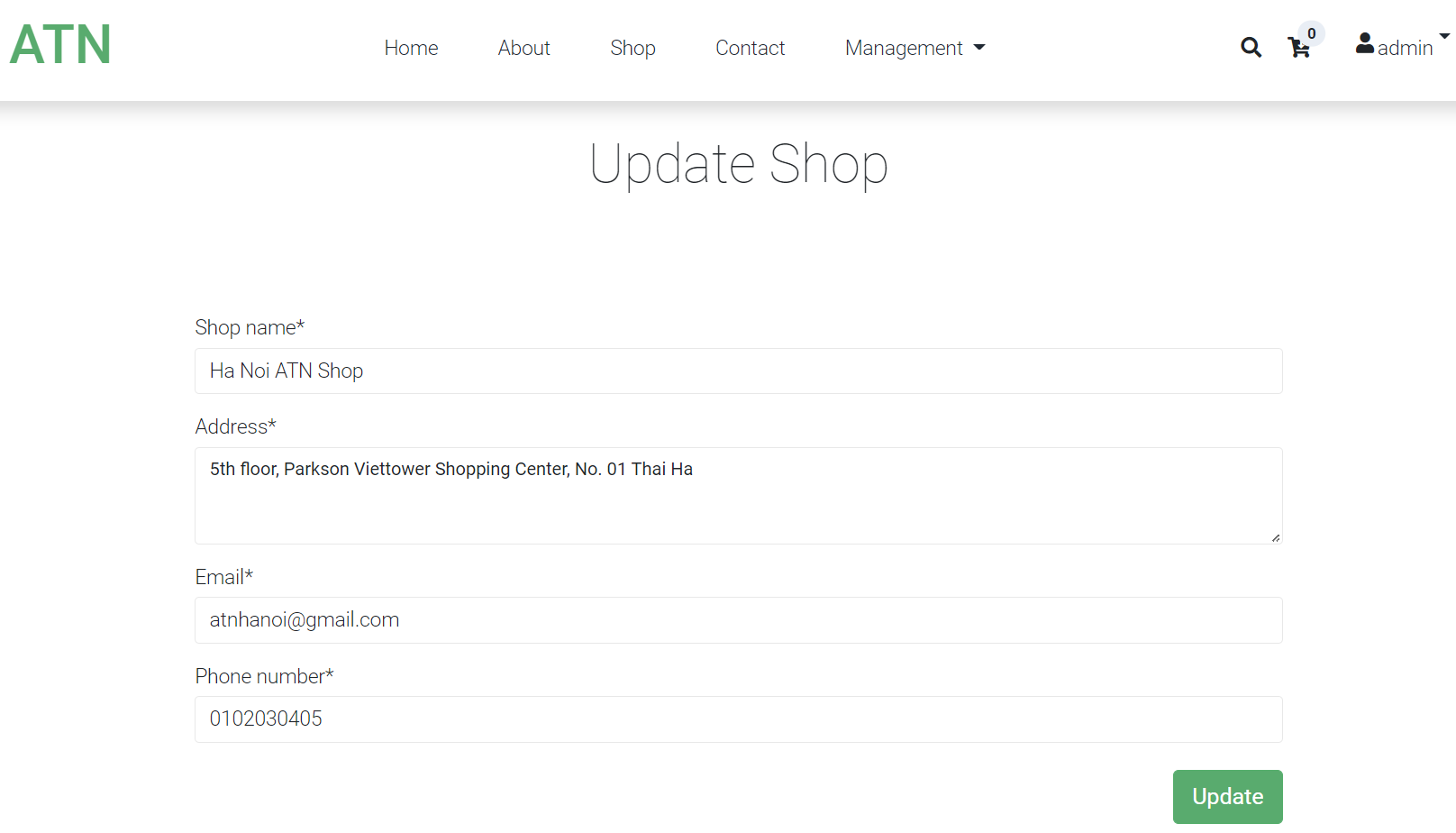
****

Figure 40: Update Shop pgae

### **2.13. Manegement Order.**

* **Manegement Order page.**

Order Manager page where all customer Orders are displayed. Admin can view, confirm order, update order information and delete order.

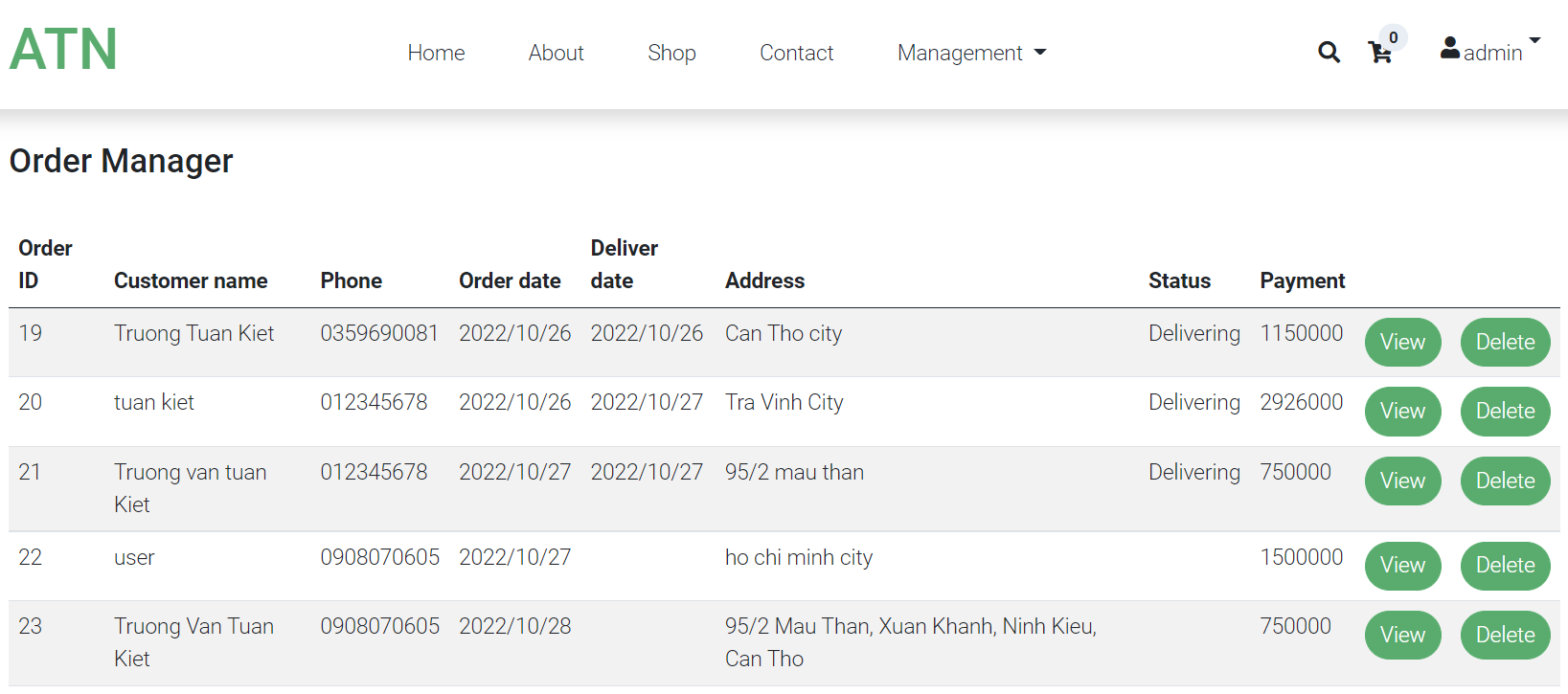
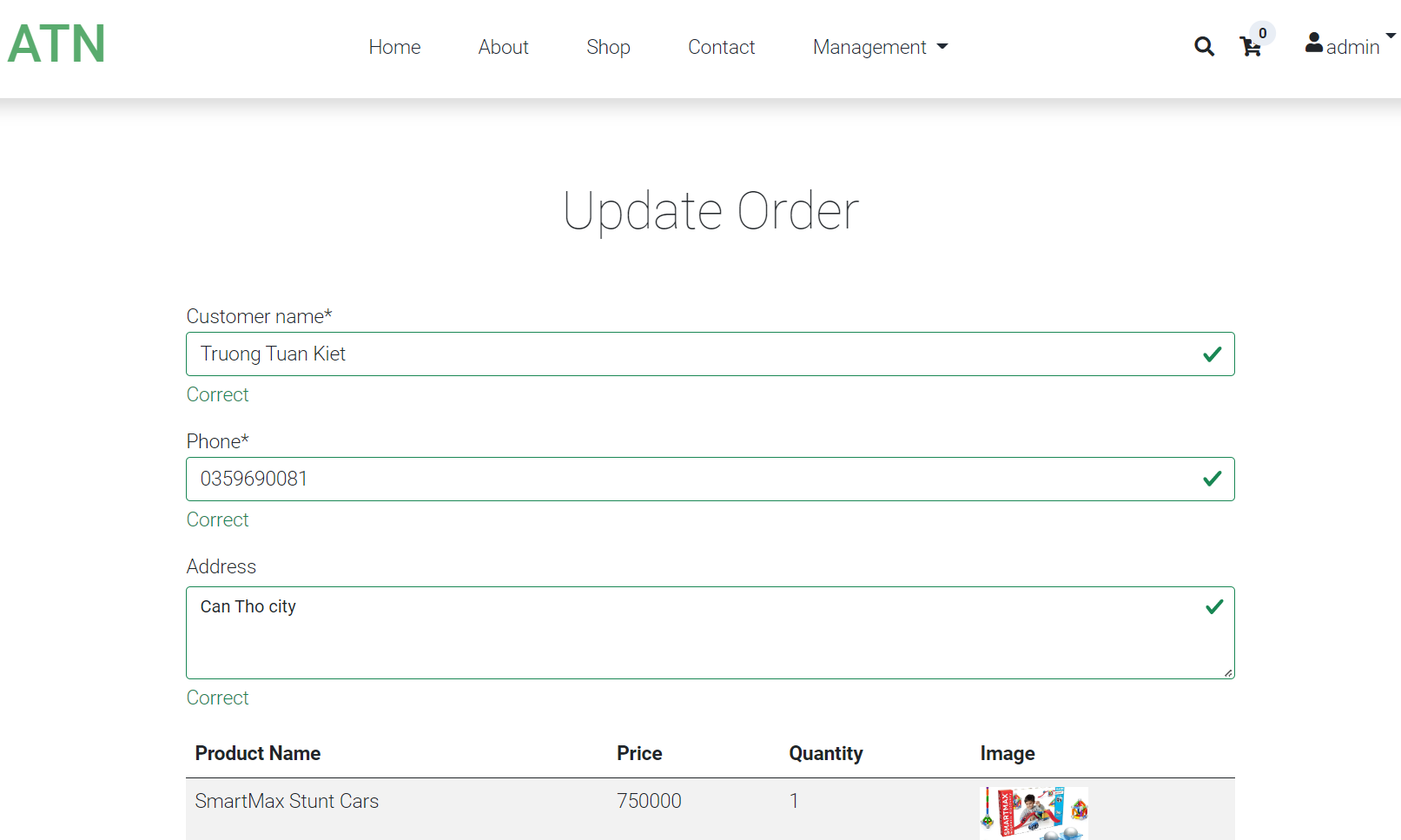
****

Figure 41: Order manager page

* **Update order page**

After clicking the "View" button on the Order Manager page Admin will go to the Update Order page. At the Update Order page, the admin can change the order status, customer information and click the "Order" button to update the order information.

****

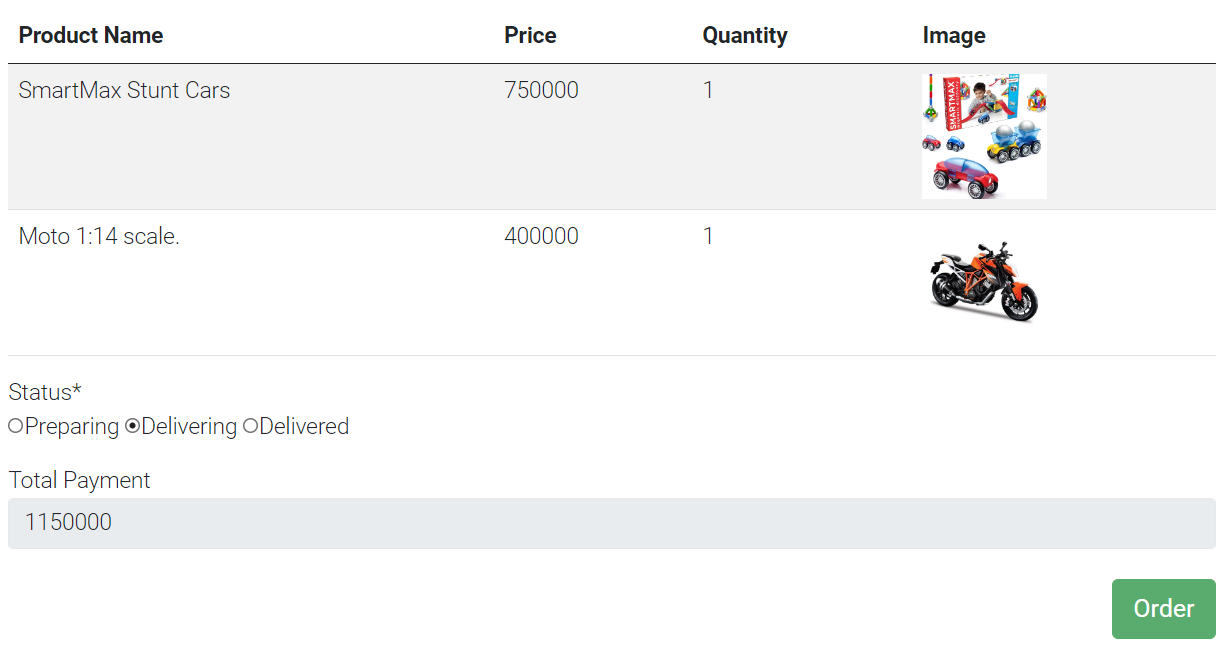
****

Figure 42: Update order page

### **2.14. Manegement shop.**

* **Shop manager page**

Shop Manager page where all the products of the system are displayed. Admin can view, add new Shop, update shop information and delete shop.

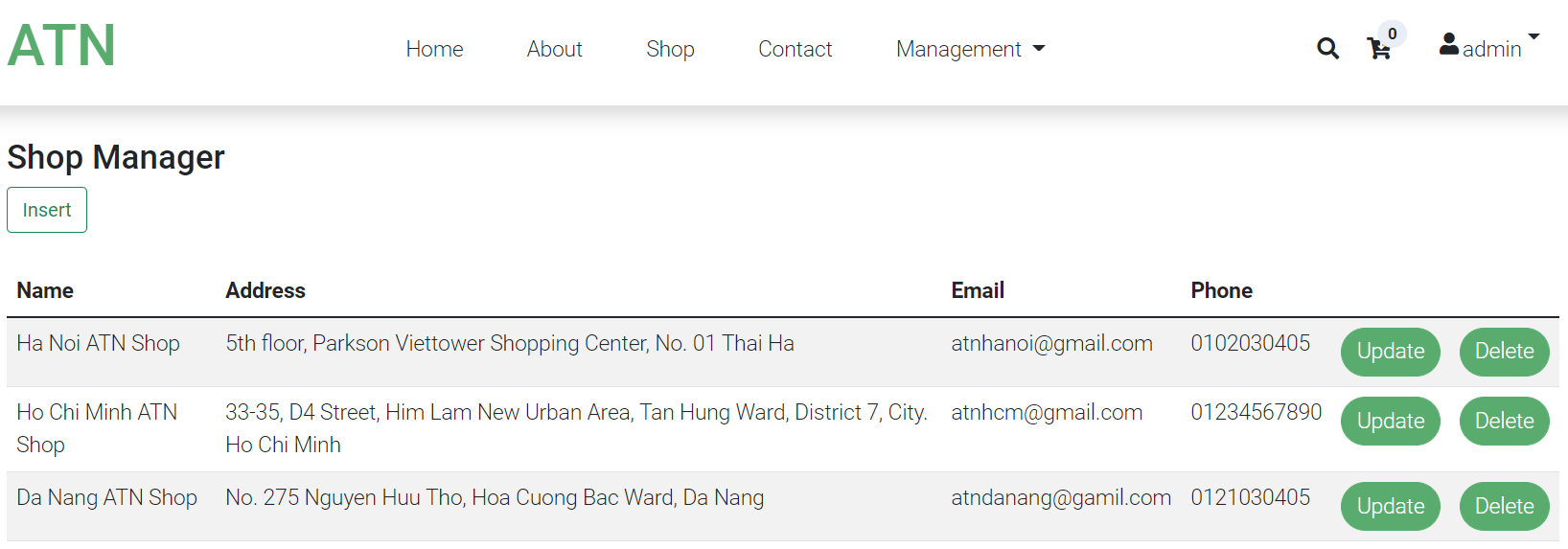
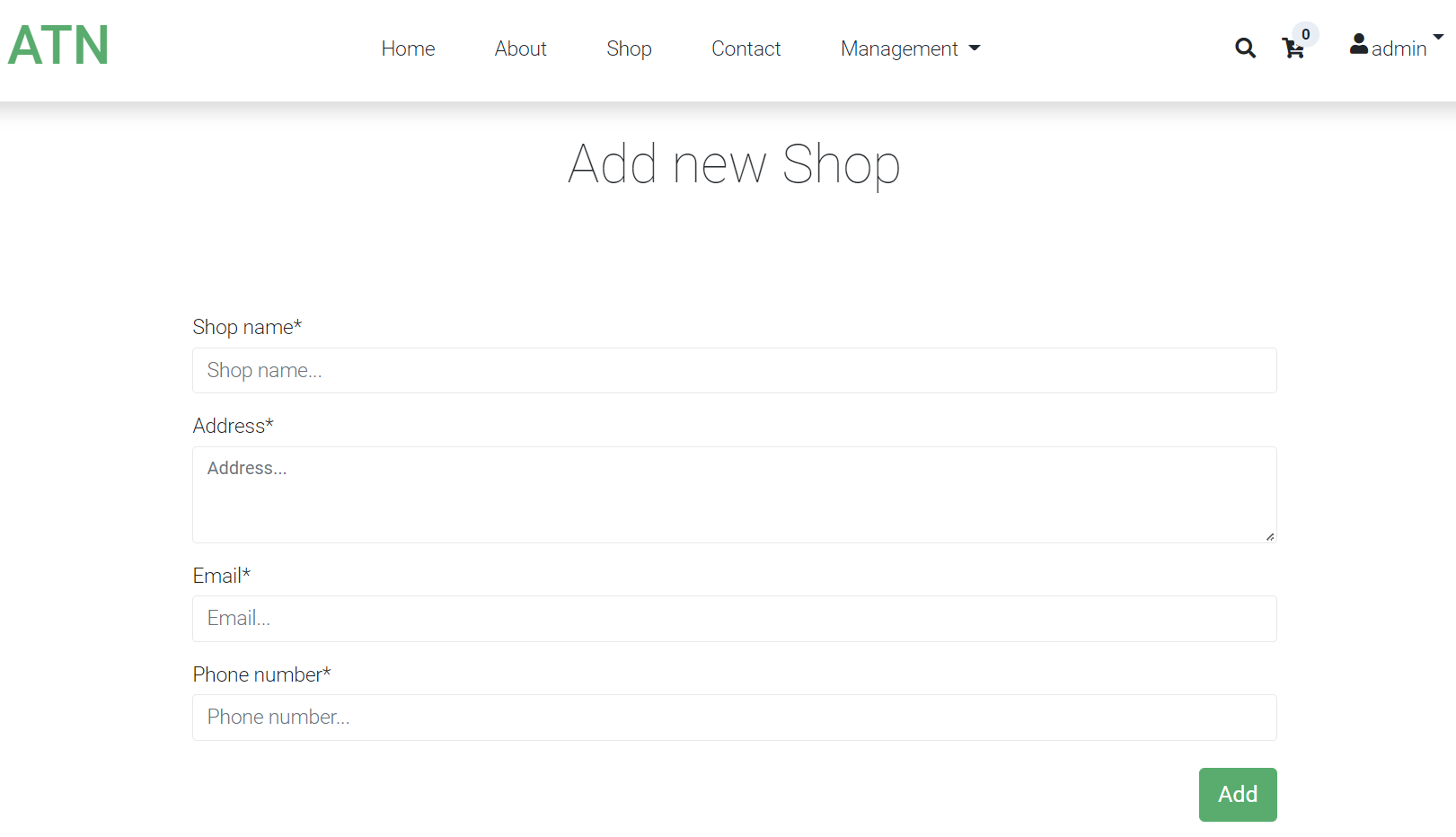
****

Figure 43: Shop manager page

* **Add new shop**

After clicking the "Inser" button on the Shop manger page, the admin will go to the "Add new shop" page. On this page, the admin can enter the shop information and press the "Add" button to add the shop to the system.

****

* **Update shop page**

After clicking the "Update" button on the Shop Manager page Admin will access the Update Shop page. At the Upadte shop page, the admin can change the shop information and click the "Upadte" button to update product information.

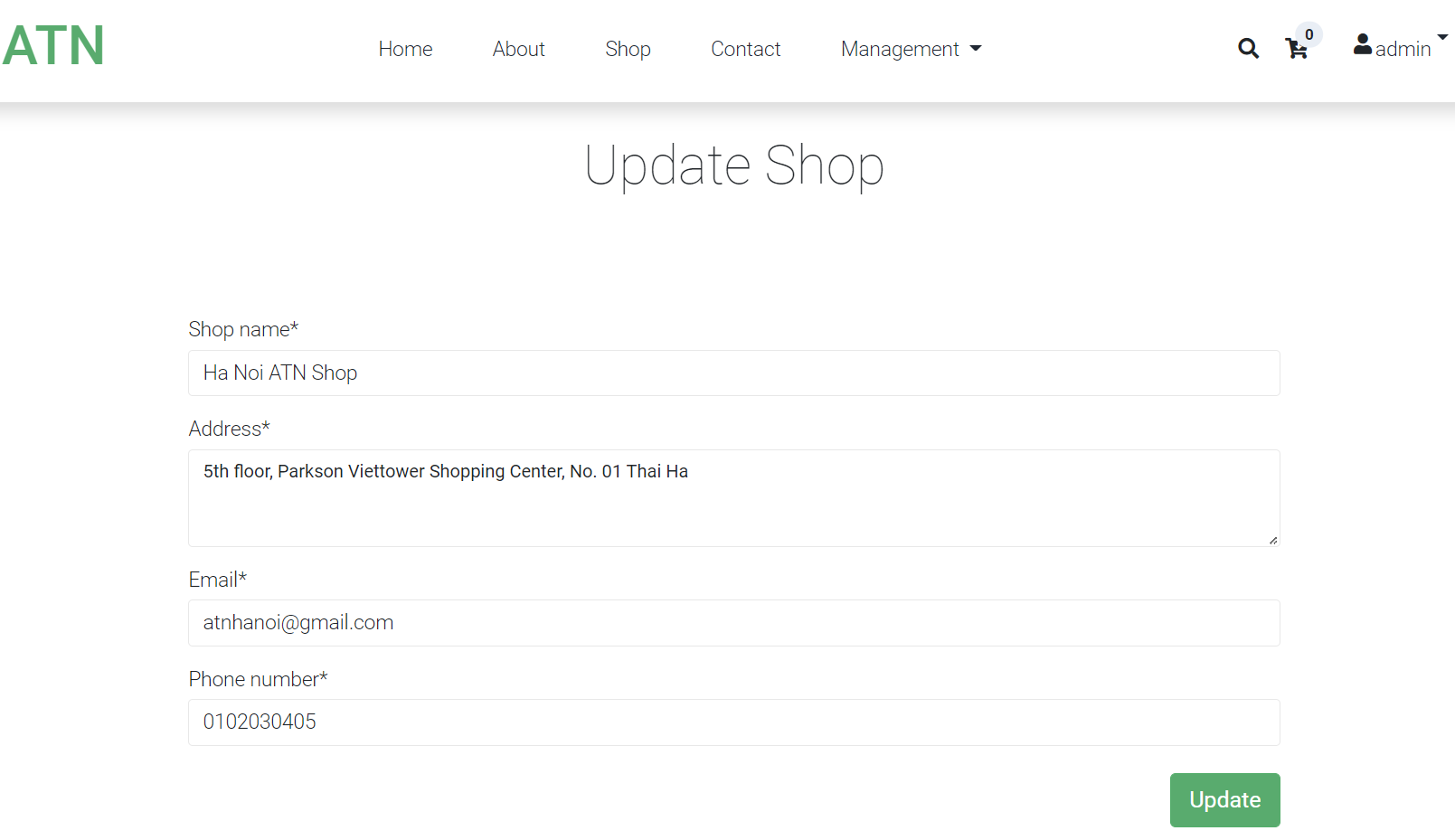
****

Figure 44: Upadte Shop

### **2.15. Manegemrnt account**

User Manager page where all the user information of the system is displayed. Admin can see all information of users registered in the system.

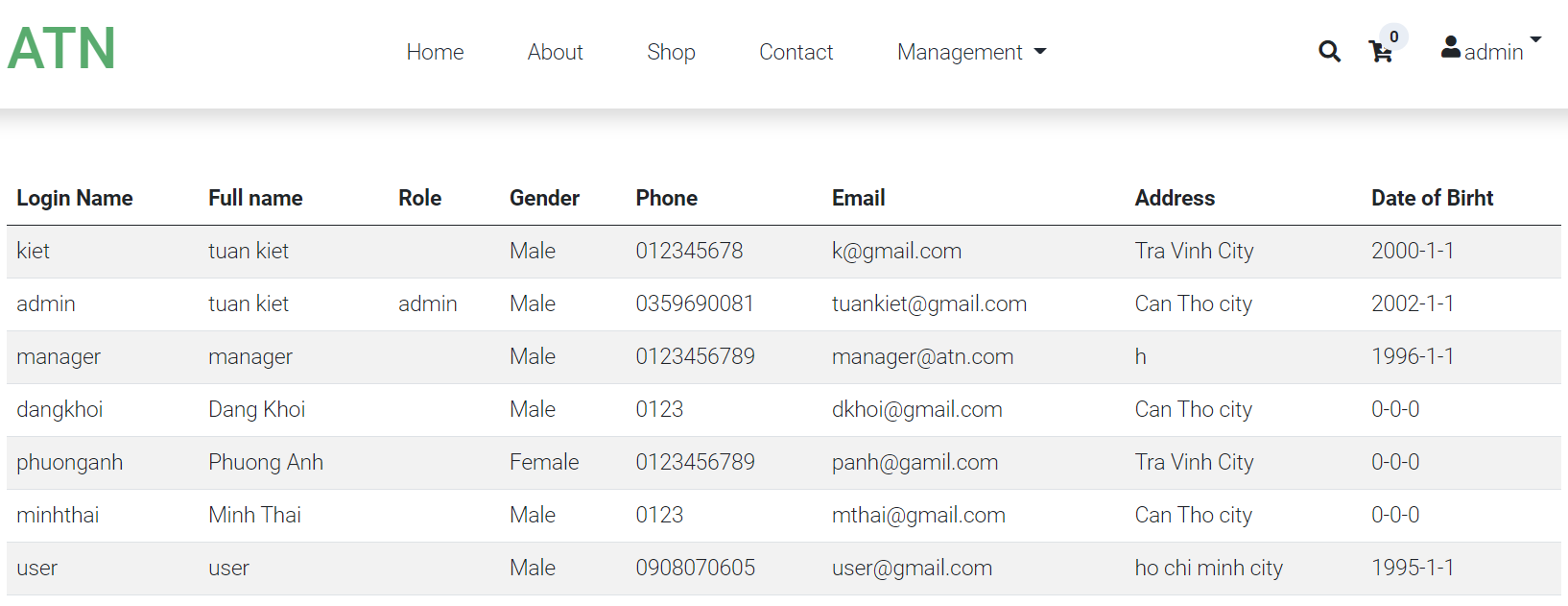
****

Figure 45: User manager page

## **3. Provide my own difficulties while developing cloud solution**

**Database problem**

I am an expert in using Microsoft SQL Server and PHP MySQL. When I switched to PostgreSQL, I had to relearn some of the PostgreSQL data types. In MySQL, when you want to declare a column using integer type, you only need to declare "int", but with PostgreSQL, you must declare "Integer". For a column to auto-increment in Microsoft SQL Server and PHP MySQL, just set "AUTO\_INCREMENT" for that column, but with PostgreSQL, you have to set the data type in that column to "SERIAL". For String data columns, in Microsoft SQL Server and PHP MySQL I just declare "varchar", but in PostgreSQL, I have to declare the type "character varying".

My solution is, find and read the website, and documentation on the PostgreSQL relational database management system such as: PostgreSQL Tutorial, W3schools, and the book Mastering PostgreSQL 13 - Fourth Edition by Hans-Jürgen Schönig.

# **II. Security**

## **1. Discuss common cloud issues and security issues**

### **1.1 Consider Basic Security Risks**

**Organizational Security Risks**

Data security is a common concern for any technology, but it becomes a major challenge when SaaS users have to rely on their providers for proper security. In SaaS, organizational data is often processed in plaintext and stored in the cloud. The SaaS provider is the one responsible for the security of the data while is being processed and stored. Also, data backup is a critical aspect in order to facilitate recovery in case of disaster, but it introduces security concerns as well. Moreover, most compliance standards do not envision compliance with regulations in a world of Cloud Computing. In the world of SaaS, the process of compliance is complex because data is located in the provider’s datacenters, which may introduce regulatory compliance issues such as data privacy, segregation, and security, that must be enforced by the provider.

**Physical Security Risks**

The physical location of the cloud data center must be secured by the CSP in order to prevent unauthorized on-site access of CSC data. Even firewalls and encryption cannot protect against the physical theft of data. Since the CSP is in charge of the physical infrastructure, they should implement and operate appropriate infrastructure controls including staff training, physical location security, network firewalls. It is also important to note that the CSP is not only responsible for storing and process data in specific jurisdictions but is also responsible for obeying the privacy regulations of those jurisdictions.

**Technological Security Risks**

These risks are the failures associated with the hardware, technologies and services provided by the CSP. In the public cloud, with its multi tenancy features, these include resource sharing isolation problems, and risks related to changing CSPs, i.e. portability. Regular maintenance and audit of infrastructure by CSP is recommended

**Compliance and Audit Risks**

These are risks related to the law. That is, risks related to lack of jurisdiction information, changes in jurisdiction, illegal clauses in the contract and ongoing legal disputes. For example, depending on location, some CSPs may be mandated by law to turn over sensitive information if demanded by government.

**Data Security Risks**

There are a variety of data security risks that we need to take into account. The three main properties that we need to ensure are data integrity, confidentiality and availability. We will go more into depth on this in the next subsection since this is the area most at risk of being compromised and hence where the bulk of cloud security efforts are focused.

These risk categories have been further split between CSPs and CSCs and are illustrated in the following two diagrams:

**Diagram

Description automatically generated**

In figure 2 we can see the five main areas of concern for a cloud service provider when it comes to security. The bullet points next to each category further narrows down a subcategory that could cause security issues to a CSP. Let us compare this to the security challenges facing the typical CSC, illustrated in figure 3.

**Diagram

Description automatically generated**

Figure 3 similarly outlines the risk categories and subcategories facing CSCs. Comparing, we can see that organizational risks are solely borne by CSPs. This is because in a cloud computing service, the infrastructure is always provided by the CSP. We can also see that both CSPs and CSCs are bound by the same considerations when it comes to compliance and audit while regarding data security, technological security and physical security, the specific concerns a CSC may have would be different from what a CSP would have. This is mainly as a result of the different levels of control they have regarding each area, as discussed in section 2. Let us now look more closely at the data security issues that need to be taken into consideration.

Having given an outline of the basic security considerations present in this subsection, we will look at the considerations that have to be made for data in the next subsection (Jain, 2015).

### **1.2. Considerations about Data Security**

**1.2.1. Data Security Properties**

**Privacy**

Privacy is one of the more important issues to deal with in the cloud and in network security in general. Privacy ensures that the personal information and identity of a CSC are not revealed to unauthorized users. This property is most important to the CSC, especially when they deal with sensitive data.

**Confidentiality**

This is related to data privacy since this is the property ensuring that the data that belongs to a CSC is not revealed to any unauthorized parties. In public clouds, the CSP is mainly responsible for securing the CSC's data. This is particularly difficult due to multi tenancy, since multiple customers have access to the same hardware that a CSC stores its data. Some providers use job scheduling and resource management, but most providers employ virtualization to maximize the use of hardware [Latif14]. These two methods allow attackers to have full access to the host and cross- VM side channel attacks to extract information from a target VM on the same machine.

**Integrity**

The integrity of data refers to the confidence that the data stored in the cloud is not altered in any way by unauthorized parties when it's being retrieved, i.e. you get out what you put in. To ensure this, CSPs must make sure that no third party has access to data in transit or data in storage. Only authorized CSCs should be able to change their data.

**Availability**

This property ensures that the CSC has access to their data, and are not denied access erroneously or due to malicious attacks by any entity. Attacks like denial-of-service are typically used to deny availability of data (Jain, 2015).

**1.2.2. Data Stages**

**Data-in-transit**

This is when data is in the process of being transmitted either to the cloud infrastructure or to the computing device used by the CSC. Here, data is most at risk of being intercepted, hence violating confidentiality. Encryption is generally used here to prevent this, along with other methods we shall detail later.

**Data-at-rest**

This is when data has been stored in the cloud infrastructure. The main issue with this stage for the CSC is their loss of control over the data. The onus of defending against attacks at this stage hence fall on the CSP. They have to ensure that all 4 of the data security properties outlined are upheld at this stage.

**Data-in-use**

This is when data is being processed into information. Here, the issues might lie with the corruption of data while it is being processed. In order to prevent this the integrity of data going into a process must be ensured using any one of the applicable methods we will discuss later

In addition to these three stages, the data left out in case of data transfer or data removal also needs to be considered, since it can cause severe security issues in the case of public cloud offerings since a CSC may end up gaining access to sections of data not properly deleted from a prior CSC.

Now that we have looked at the various issues that need to be guarded against in cloud computing, we can see that by far the largest and most complex area is that of data security. We shall hence focus slightly more on the methods used in industry to preserve data security, and give an overview of the more straightforward techniques used in the other risk areas (Jain, 2015).

## **2. Provide solutions of some issues which I can**

### **2.1. Countermeasures for Security Risks**

**Organizational Security Risks**

Malicious Insiders - The risk of having malicious personnel in a CSPs staff can be mitigated by putting strict legal constraints in contracts when hiring personnel. A comprehensive assessment of the CSP by a third party, as well as a robust security breach notification process

**Physical Security Risks**

Physical Breach - The risk of intruders gaining physical access to devices used in the provision of cloud services can be reduced by having strong physical security deterrents in place such as armed guards, keycard access and biometric scans to restrict access to sensitive locations in the data center.

**Technological Security Risks**

Secure virtualization - CSP can use an Advanced Cloud Protection system (ACPS) to ensure the security of guest virtual machines and of distributed computing middleware. Behaviour of cloud components can also be monitored by logging and periodic checking of executable system files.

 There should be separate domains for providers and users, each with a special trust agent.

**Compliance and Audit Risks**

Both CSPs and CSCs need to understand legal and regulatory obligations and ensure that any contracts made meet these obligations The CSP should also ensure that its discovery capabilities do not compromise security and privacy of data (Jain, 2015).

### **2.1. Methods to ensure Data security**

**Authentication in the Cloud**

 identity and access management (IAM). IAM ensures regulatory compliance by managing the major security concerns - authentication, automated provisioning and authorization services. Other underlying technologies used for authentication, authorization and access control services are OpenID, OAuth, SAML, XACML.The trusted computing group's (TCG's) IF-MAP standard further allows for real-time communication between a cloud service provider and the customer about authorized users and other security issues.

Authentication for the CPC can be done either by the CSP or outsourced to third party specialists. Some methods for authentication include the identity-based hierarchical model for cloud computing (IBHMCC) and the SSH Authentication Protocol (SAP)[Spoorthy14] . This is used mainly to protect data privacy and confidentiality.

**Encryption techniques in the cloud**

Caesar Cipher: It is a classical substitution cipher. A simple example of such a cipher replaces the letter of alphabet with a letter that is 3 paces ahead of it, for example "ZULU" will be converted into "CXOX". There are only 25 possible key options and as such this cipher can easily be brute forced. Caesar Cipher: It is a classical substitution cipher. A simple example of such a cipher replaces the letter of alphabet with a letter that is 3 paces ahead of it, for example "ZULU" will be converted into "CXOX". There are only 25 possible key options and as such this cipher can easily be brute forced.

S-DES - Simplified Data Encryption Standard has a process of key generation where instead of using a key as is for encryption and decryption, the key generation process of S-DES generates 2 sub keys after processing the initial 10 bit input. The two sub keys are generated at both the transmission and receiving ends. With the inclusion of initial permutation and expansion permutations the security is substantial when compared with the classical techniques, S-DES gives some structure and formation to encryption techniques with step to step procedures for both encryption and decryption.

RSA - A cryptographic algorithm whose encryption key is public and differs from the decryption key which is kept secret. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, the creators of the algorithm. This algorithm is based on the fact that finding the factors of an integer is hard. It is one of the more commonly used encryption algorithms nowadays.

Secure Socket Layer (SSL) 128 bit encryption - it is a commonly-used protocol for managing the security of a message transmission on the Internet and it uses public and private key encryption system (Jain, 2015).

## **3. Discuss how to overcome these security issues when building a secure cloud platform**

### **3.1. How to Safeguard Your Data from Cloud Misconfigurations**

According to (Sukianto, 2022), the following expert recommendations will help you securely configure your cloud environment and maintain its security:

**Remember forgotten services** – Development and operations teams mostly create new cloud applications and servers, configure them, then fail to recheck the configuration. Ensure you know where your cloud services and assets are and their status.

**Develop policy and templates** – IT leaders must propagate working security settings into their environments' base configuration settings to allow future instances of a piece of cloud infrastructure or application to benefit from past lessons.

**Automate security and configuration checks** – Agile development approaches leverage extensive automation to create and deploy secure code. Therefore, make sure you check your running infrastructure and applications for security and compliance. Automation can be beneficial here.

**Leverage provider tools** – You must understand the extent to which you share your security responsibility with the cloud provider. More responsibility is on the customer's end with infrastructure-as-a-service clouds, while the cloud service provider primarily manages SaaS offerings.

**Conduct risk assessments** – Cybersecurity risk assessments help you identify potential threats in your cloud storage and other infrastructure sections when migrating your data and operations to the cloud.

### **3.2. The Ways to Prevent Data Leaks in the Cloud**

According to (Jaylin, 2021), here are the good ways to make sure that your cloud data is secure:

**Use good quality encryption, both on stored data and connections.** Avoid sending sensitive information “in the clear.” Avoid using public and hotel wi-fi. If you travel frequently for work you should set up a VPN which will encrypt your data regardless of how you are connected.

**Educate employees about phishing.** The easiest way to access cloud data is to trick somebody out of their username and password. Make sure all employees know that they should not click on links in email unless they know who sent it and were expecting the email. Some phishing emails can look very authentic. It can be worth having IT periodically send out phishing emails to see who is fooled by them, targeting individuals for extra training.

**Use two-factor authentication.** Although this can be a pain for users, it means that if somebody’s username and password are stolen, the criminals will still have to do a lot more work to get into their account. This could be as simple as asking security questions or it could be requiring somebody to use a verification code sent to their phone. Device-based authentication is generally the most secure.

**Have a decent password policy.** Traditional password policies often create passwords that are hard for humans to remember and easy for computers to guess. Educate employees on using pass phrases and other systems that are harder to hack. Encourage the use of a password manager to allow for more complicated passwords with less risk of forgetting (or of people storing passwords in a less secure location because they can’t remember them. Also make sure that passwords expire every 90 days or so. Some systems will not allow the same password to be repeated within two years or so.

**Set correct user permissions.** If somebody is terminated, you should close down their access immediately (ideally at the exact time you tell them so they cannot do damage on their way out). Do not give people access to data they don’t need access to. It’s not about not trusting them, but about minimizing the damage compromising one account can do. Also, accidents happen. We have all seen that private email that gets sent to the entire company network. With modern computer systems auto completing emails, it’s easy for sensitive data to be sent to the wrong party.

**Keep backups.** It’s worth keeping an extra backup outside your normal cloud provider, especially for the most important data. It can also be worth keeping at least some data on a physical drive that is protected by site security. Good backups can protect you from ransomware and similar attacks or from data destruction, intentional or otherwise.

**Use the right cloud provider.** Have a conversation with your cloud provider about their security methods. How do they back up your data? Do they have site security to prevent physical access to the servers? A good cloud provider will care as much about the security of your data as you do, especially as their reputation hinges on it.

### **3.3. Cloud Computing Security Services to Mitigate DDoS Attacks**

The focus of this research is on distributed denial of service (DDoS) attacks on the cloud. The authors researched on existing cloud security solutions and also present an implementable solution focusing on DDoS mitigation for IT infrastructure. The authors define the scope and recommend few focus areas:

* Defending volumetric attacks is a need for cloud components.
* Blocking application-level attacks without submitting SSL Key.
* Deploying acceptable network infrastructure as per IT security policy.

DDoS attack mitigation solutions are discussed here based on design perspective:

**3.3.1 On-premise based:**

Having a devoted on-premise DDoS attack mitigation answer are first-rate desirable for government entities, financial establishments, and healthcare but not beneficial for all. When the highest stage of safety is mandatory and organizations opt to give as little visibility into their customer facts or approximately their encryption certificate to as few third birthday celebration providers, this could be regarded as a limited scope option. On-premise DDoS devices might store encryption certificates and inspect visitors regionally without any scrubbing, redirection, or inspection. The mitigation device would be required to guard against numerous DDoS vectors like flooding (UDP/ICMP, SYN), SSL based, application layer (HTTP GET/POST), or low and slow attacks. With mitigation structures in house, the proximity to facts center sources is useful, and the systems may be fine-tuned at once by the in-residence IT teams. They have a tendency to have a miles more cognizance to their setup for any adjustments in site visitor flows or from the application servers. Thus, they might have a tendency to have a higher chance of detecting any suspicious traits or visitors requests.

**3.3.2 Cloud-based security services:**

In providing anti-DDoS and superior mitigation protection in shape of managed security services, many cloud carrier companies offer protection from community floods with the aid of deploying mitigation system on the ISP network edge stage or with scrubbing centers. This involves traffic diversion from the corporation network to detection or scrubbing center. When a DDoS attack starts, human intervention is needed and takes as a minimum of 15–30 minutes all through which the online services are left unprotected and exposed. The cloud-based totally DDoS mitigation service guarantees quantity blocking off of community flood assaults from accomplishing the corporation edge devices or flooding the WAN circuit which is free of volumetric community flood attack. However, there exist glaring problems with a cloud primarily based on DDoS mitigation offerings.

* Cannot discover and block application layer attacks and slow attack.
* Unable to defend stateful infrastructure structures like firewalls or IPS.
* Unable to deal with attacks like software layer attack, state exhaustion, and multi-vector attacks.

**3.3.3. Hybrid cloud-based security:**

Using hybrid cloud functions gives the best-of-breed mitigation option, where the hybrid infrastructure combines the on-premise in-house setup with DDoS mitigation carriers to act as an included mitigation solution. In hybrid solutions, another option is to use a devoted DDoS mitigation provider’s capability in order to detect and block a couple of DDoS vectors. Having public cloud issuer dynamically booms the community pipe bandwidth for the duration of a DDoS attack; takes off a while after being detected, till the time mitigation begins; and saves the on-premise infrastructure from the attack and affecting the provision of its online services. Typical answer is in the course of DDoS attack; the entire site visitors are diverted to a DDoS mitigation issuer’s cloud, where it is scanned, scrubbed with the attack visitors getting diagnosed, and removed before being re-routed lower back to the in-residence information middle of the enterprise. Hybrid solutions permits organizations to gain from the following:

* Widest security coverage that can simplest be finished by means of combining on-premise and cloud insurance.
* Shortest reaction time by using an on-premise solution that begins right away and mechanically to mitigate the assault.
* Single touch point during an attack both for on-premise and cloud mitigation.
* Scalability—each tier is impartial of the other and can scale horizontally, in case there is a web application attack spike, adding extra WAF devices to ensure enough WAF capability may be done within the application defense tier without affecting the community tier.
* Performance—on the grounds that requests come in tiers, network utilization is minimized, and load decreased overall.
* Availability—with hybrid solutions, if the first or second tier is down, at least there is one tier left to serve consumer requests. This satisfies the BCP of the organisation.
* Vendor independence—community and application protection infrastructure can setup the usage of hardware structures or even specific software program versions.
* Policy independence—while new policies are implemented at the application defense tier, the opposite tier directs simplest that specific visitors in the direction of the rules until they are established and ready for production use.

# **References**

Jain, P. R., 2015. *Recent Advances in Networking (Data Center Virtualization, SDN, Big Data, Internet of Things).* [Online]   
Available at: https://www.cse.wustl.edu/~jain/cse570-15/index.html  
[Accessed 2022].

Jaylin, B., 2021. *7 Ways to Prevent Data Leaks in the Cloud.* [Online]   
Available at: https://www.otava.com/blog/7-ways-to-prevent-data-leaks-in-the-cloud/  
[Accessed 2022].

Sukianto, A., 2022. *Common Cloud Misconfigurations and How to Avoid Them.* [Online]   
Available at: https://www.upguard.com/blog/cloud-misconfiguration  
[Accessed 2020].