



E-commerce APPLICATION IN CLOUD FOUNDARY using cloud application development



A PROJECT REPORT

Submitted by

Anirish. S. V	611821106004
Dhanush. S	611821106004
Magesh. T	611821106029
Gokul. P	611821106016
Saran. K	611821106052

BACHELOR OF ENGINEERING

in

DEPARTMENT OF ELECTRONICS AND
COMMUNICATION ENGINEERING

P.S.V. COLLEGE OF ENGINEERING AND TECHNOLOGY
(Accredited by the NAAC with 'A' Grade)
(An ISO 9001:2015 Certified Institution)
KRISHNAGIRI

ANNA UNIVERSITY-CHENNAI 600 025
MAY 2023

E-COMMERCE APPLICATION ON IBM CLOUD FOUNDRY

DESIGN OF E-COMMERCE APPLICATION ON IBM CLOUD FOUNDRY IN

JAVASCRIPT

Creating an innovative e-commerce application on IBM Cloud Foundry can be an exciting project. IBM Cloud Foundry provides a platform-as-a-service (PaaS) environment that allows you to build and deploy applications easily. Here's a step-by-step guide for your innovation project:

1. **Project Scope and Objectives:** Define the goals of your e-commerce application. What problems will it solve? Who is your target audience? What features will it have that differentiate it from existing e-commerce platforms?
2. **Market Research:** Conduct market research to understand the competition and customer needs. Identify gaps in the market that your application can address.
3. **Conceptualization and Design:** Create wireframes and mockups of your application's user interface. Design the user experience (UX) to be intuitive and engaging. Consider how users will navigate through your application.
4. **Choose IBM Cloud Services:** Decide which IBM Cloud services you'll use for your application. For an e-commerce platform, you may need services like IBM Cloud Databases, IBM Cloud Object Storage, and IBM Watson for AI-powered features.
5. **Development:** Start building your application using the chosen technology stack. IBM Cloud Foundry supports multiple programming languages, so choose the one that suits your team's expertise.
6. **Database Integration:** Implement the database to store product information, user profiles, and order history. Consider using IBM Cloud Databases for scalability and reliability.
7. **Security:** Implement robust security measures, including encryption, secure authentication, and authorization mechanisms to protect user data and transactions.
8. **Payment Gateway Integration:** Integrate a secure payment gateway to handle transactions. IBM Cloud supports various payment gateway providers.
9. **Machine Learning and AI:** Utilize IBM Watson or other AI services for personalized product recommendations, chatbots, or fraud detection.
10. **Testing:** Thoroughly test your application to identify and fix bugs and vulnerabilities. Perform load testing to ensure it can handle traffic spikes.
11. **Deployment:** Deploy your application on IBM Cloud Foundry. Configure scaling options to accommodate traffic fluctuations.

12. **Monitoring and Analytics:** Set up monitoring tools and analytics to track user behavior, performance, and application health. Use IBM Cloud Monitoring and IBM Cloud Log Analysis for this purpose.
13. **User Testing:** Conduct user testing to gather feedback and make necessary improvements.
14. **Marketing and Promotion:** Develop a marketing strategy to promote your e-commerce application. Use social media, content marketing, and other strategies to reach your target audience.
15. **Feedback Loop:** Continuously gather user feedback and make iterative improvements to enhance your application.
16. **Maintenance and Support:** Provide ongoing maintenance and support for your application. Address issues promptly and keep the platform up to date with security patches and new features.
17. **Scaling and Growth:** Plan for the scalability of your application as it grows. Use IBM Cloud's auto-scaling features to handle increased traffic.
18. **Compliance and Regulations:** Ensure your e-commerce application complies with data protection and e-commerce regulations in your target markets.
19. **Documentation:** Create comprehensive documentation for developers, administrators, and users.
20. **Feedback and Evaluation:** Continuously gather feedback from users and stakeholders and evaluate the success of your e-commerce application against your initial objectives.

Remember that innovation is an ongoing process. Stay updated with emerging technologies and customer trends to keep your e-commerce application competitive and innovative.

Step 21:

```
const express = require('express');
const app = express();
const port = process.env.PORT || 3000;

// Middleware for handling JSON data
app.use(express.json());

// Routes
app.get('/', (req, res) => {
```

```
res.send('Welcome to the e-commerce application!');

});
```

```
// Define routes for products, users, and orders here
```

```
app.listen(port, () => {
  console.log(`Server is running on port ${port}`);
});
```

E-commerce application on IBM Cloud

Boundary

This example provides a basic structure. You'd typically use a front-end framework for more complex user interfaces and a real database for storing product data. Additionally, security features like user authentication, payment processing, and data validation are essential for a production e-commerce application.

To deploy this on IBM Cloud Foundry, you would need to:

1. Create an IBM Cloud account and set up your Cloud Foundry space.
2. Install the IBM Cloud CLI and log in to your account.
3. Push your code to IBM Cloud Foundry using the `cf push` command.

Remember that a production-ready e-commerce application involves various security and performance considerations and is usually developed by a team of professionals. This example is just a starting point for educational purposes.

Continuing from the previous response, here's a more detailed overview of developing an e-commerce application on IBM Cloud Foundry:

1. **Architecture Design**:

- Plan your application's architecture, considering components like the front-end, back-end, and databases.
- Choose appropriate programming languages and frameworks based on your specific requirements.

2. **Front-End Development**:

- Create the user interface for your e-commerce website using HTML, CSS, and JavaScript.
- Utilize front-end frameworks like React, Angular, or Vue.js for interactive and responsive designs.

3. **Back-End Development**:

- Develop the back-end of your application to handle business logic, data processing, and integration with databases.
- Use server-side languages like Node.js, Java, Python, or Ruby based on your expertise and project requirements.

4. **Database Integration**:

- Set up a database to store product information, customer data, and order history. IBM Cloud offers databases like Db2 and Cloudant that you can use.

5. **Security Implementation**:

- Implement security measures, including encryption, authentication, and authorization, to protect user data and transactions.

6. **Payment Integration**:

- Integrate with payment gateways and ensure secure handling of transactions. Popular options include Stripe, PayPal, or IBM Payment Gateway.

7. **Order Processing**:

- Develop features for order placement, processing, and order status tracking.

8. **User Authentication and Authorization**:

- Implement user registration and login functionality, with role-based access control to manage user privileges.

9. **Search and Navigation**:

- Incorporate search functionality and navigation features to help users find products easily.

10. **Analytics and Reporting**:

- Set up analytics tools or services to gather data on user behavior, sales, and website performance.

11. **Testing and Quality Assurance**:

- Thoroughly test your application to ensure it's bug-free, responsive, and performs well under various conditions.

12. **Deployment on IBM Cloud Foundry**:

- Utilize IBM Cloud Foundry to deploy your application. IBM provides documentation and tools to make this process smooth.

13. **Scalability and Load Balancing**:

- Configure auto-scaling and load balancing to handle traffic spikes during promotions or high-demand periods.

14. **Monitoring and Maintenance**:

- Implement monitoring and alerting systems to keep an eye on your application's health and performance. IBM Cloud offers monitoring services for this purpose.

15. **Continuous Integration/Continuous Deployment (CI/CD)**:

- Set up CI/CD pipelines for automated testing and deployment, ensuring smooth updates and releases.

16. **Compliance and Regulations**:

- Ensure your e-commerce application complies with data protection regulations, like GDPR or CCPA.

17. **Customer Support and Feedback**:

- Integrate customer support features and gather user feedback to improve your application continually.

Remember that developing an e-commerce application is an ongoing process, and you'll need to adapt to changing market trends and customer needs. IBM Cloud Foundry offers a robust platform for hosting and scaling your e-commerce application.

Creating a full e-commerce application on IBM Cloud Foundry is a complex task, and it involves a lot of code across various components like front-end, back-end, and databases. Below, I'll provide a simplified example of creating a basic product listing page using Node.js for the back-end and a simple HTML page for the front-end. This code is for educational purposes and should be extended and secured for a real e-commerce application.

Back-End (Node.js) - Create a basic server and API for product listing:

```
```javascript
// app.js

const express = require('express');
const app = express();
const port = process.env.PORT || 3000;

const products = [
 { id: 1, name: 'Product 1', price: 10.99 },
 { id: 2, name: 'Product 2', price: 15.99 },
 // Add more products here
];

app.get('/api/products', (req, res) => {
 res.json(products);
});

app.listen(port, () => {
 console.log(`Server is running on port ${port}`);
});
```

```

****Front-End (HTML)**** - Create a simple HTML page to display the product listings:

```
```html
<!-- index.html -->

<!DOCTYPE html>
<html>
<head>
<title>E-commerce Store</title>
</head>
<body>
<h1>Products</h1>
<ul id="product-list">

<script>
fetch('/api/products')
.then(response => response.json())
.then(products => {
 const productList = document.getElementById('product-list');
 products.forEach(product => {
 const listItem = document.createElement('li');
 listItem.textContent = `${product.name} - ${product.price}`;
 productList.appendChild(listItem);
 });
})
.catch(error => {
 console.error('Error fetching products:', error);
});
</script>
```

```
</body>
</html>
```
```

This example provides a basic structure. You'd typically use a front-end framework for more complex user interfaces and a real database for storing product data. Additionally, security features like user authentication, payment processing, and data validation are essential for a production e-commerce application.

To deploy this on IBM Cloud Foundry, you would need to:

1. Create an IBM Cloud account and set up your Cloud Foundry space.
2. Install the IBM Cloud CLI and log in to your account.
3. Push your code to IBM Cloud Foundry using the `cf push` command.

Remember that a production-ready e-commerce application involves various security and performance considerations and is usually developed by a team of professionals. This example is just a starting point for educational purposes.

E-XOMMERCE APPLICATION ON IBM CLOUD

FOUNDARY

Creating an e-commerce application on Cloud Foundry involves several steps. Here's an overview of the process:

1. **Preparation**:

- **Plan Your Application**: Define your e-commerce application's requirements, such as the technology stack, database, and third-party services it needs.
- **Set Up a Cloud Foundry Account**: Choose a Cloud Foundry provider (e.g., IBM Cloud, Pivotal Web Services) and create an account if you don't have one.

2. **Develop Your E-commerce Application**:

- Write your application code and set up your database, ensuring it's compatible with Cloud Foundry's supported technologies.
- Containerize your application if desired using tools like Docker.

3. **Install Cloud Foundry Command Line Tools**:

- Download and install the Cloud Foundry Command Line Interface (CLI) tools. These tools allow you to interact with Cloud Foundry from your terminal.

4. **Login to Cloud Foundry**:

- Use the `cf login` command to authenticate your account with the Cloud Foundry provider. You'll need to provide your credentials.

5. **Push Your Application**:

- Use the `cf push` command to deploy your e-commerce application to Cloud Foundry. This command will upload your code and create a new application instance.

6. **Configure Services**:

- If your e-commerce application relies on databases, caching, or other services, you can use Cloud Foundry's marketplace to create and bind these services to your application. For example, you can bind a PostgreSQL database service to your app.

7. ****Scale Your Application**:**

- Use the `cf scale` command to adjust the number of instances (horizontal scaling) and resources allocated to your application based on your traffic requirements.

8. ****Manage Environment Variables**:**

- Store configuration details, API keys, and other sensitive information in environment variables within Cloud Foundry to enhance security.

9. ****Domain Setup**:**

- Map your e-commerce application to a custom domain or subdomain if needed. This can usually be done through the Cloud Foundry provider's control panel or CLI.

10. ****Monitoring and Logs**:**

- Use Cloud Foundry's built-in tools to monitor the performance of your application and access logs for debugging and auditing.

11. ****Security and Compliance**:**

- Implement security best practices, including data encryption, access controls, and regular security updates to protect sensitive customer data.

12. ****Backup and Recovery**:**

- Set up regular backups of your application's data and implement a disaster recovery plan to ensure business continuity.

13. ****Testing and Optimization**:**

- Conduct thorough testing, including load testing, and continuously optimize your application's performance and user experience.

14. ****Scaling and Auto-Healing**:**

- Implement auto-scaling rules and health checks to ensure your application can automatically handle traffic spikes and recover from failures.

15. ****Regular Updates**:**

- Keep your e-commerce application and its dependencies up-to-date to benefit from security patches and new features.

Remember that the specific steps and commands may vary slightly depending on the Cloud Foundry provider and the technology stack you're using for your e-commerce application. Additionally, it's important to regularly monitor and maintain your application to ensure it runs smoothly and securely on Cloud Foundry.

Creating an entire e-commerce application on Cloud Foundry is a complex task that involves a lot of code and configuration. It would not be feasible to provide the entire code for such an application here. However, I can provide you with a high-level overview and some code snippets for essential components:

1. ****Backend Server**:**

- You would need to develop the backend server to handle product listings, user accounts, shopping carts, and order processing. Popular technologies include Node.js, Ruby on Rails, Python (Django), or Java (Spring Boot).

Here's a simple example using Node.js with Express for a basic server:

```
```javascript
const express = require('express');
const app = express();
const port = process.env.PORT || 3000;

// Define routes for product listings, user accounts, etc.

app.listen(port, () => {
 console.log(`Server is running on port ${port}`);
})
```

```
});
```

```
...
```

## 2. \*\*Frontend\*\*:

- Develop the user interface using HTML, CSS, and JavaScript (e.g., React, Angular, or Vue.js). Your frontend code would handle product display, user registration, shopping cart functionality, and payment processing.

A simple example using HTML and JavaScript:

```
```html
<!-- HTML for displaying a product -->
<div class="product">
  <h2>Product Name</h2>
  <p>Price: $10</p>
  <button onclick="addToCart(1)">Add to Cart</button>
</div>

<script>
function addToCart(productId) {
  // Add the selected product to the shopping cart
}
</script>
...```

```

3. **Database**:

- Use a database system (e.g., PostgreSQL, MySQL) to store product information, user data, and order history.

4. **Cloud Foundry Deployment**:

- Use the Cloud Foundry CLI to deploy your application to the Cloud Foundry platform. Here's an example command for deploying a Node.js app:

```
```shell
cf push your-app-name -p path-to-your-app
```

```

5. ****Payment Integration**:**

- Integrate with payment gateways such as Stripe or PayPal for processing customer payments.

6. ****User Authentication**:**

- Implement user authentication and authorization to secure user accounts and protect sensitive information.

7. ****Security**:**

- Implement security practices to protect user data, including encryption, input validation, and protection against common web vulnerabilities.

8. ****APIs**:**

- If you plan to integrate with third-party services or marketplaces, develop and consume APIs as needed.

9. ****Testing**:**

- Write unit tests and conduct end-to-end testing to ensure the reliability of your e-commerce application.

Please note that developing a complete e-commerce application is a significant project that involves multiple components and functionalities. You may need a team of developers, designers, and testers to build and maintain such an application. The code examples provided here are very basic and serve as a starting point. Building a production-ready e-commerce application requires careful planning and implementation.

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/329417336>

Cloud Computing of E-commerce

Article in *Modern Applied Science* · January 2019

DOI: 10.5539/mas.v13n1p27

CITATIONS

17

READS

25,262

2 authors:



Tamara Almarabeh

University of Jordan

24 PUBLICATIONS 869 CITATIONS

[SEE PROFILE](#)



Yousef Majdalawi

University of Jordan

15 PUBLICATIONS 291 CITATIONS

[SEE PROFILE](#)

Cloud Computing of E-commerce

Tamara Almarabeh¹ & Yousef Kh. Majdalawi¹

¹ Department of Computer Information Systems, The University of Jordan, Jordan

Correspondence: Tamara Almarabeh, Department of Computer Information Systems, The University of Jordan, Jordan. E-mail: t.almaraabeh@ju.edu.jo

Received: October 28, 2018

Accepted: November 8, 2018

Online Published: December 5, 2018

doi:10.5539/mas.v13n1p27

URL: <https://doi.org/10.5539/mas.v13n1p27>

Abstract

Cloud computing affects on different sectors, including: E-learning, health care, and E-commerce. It offers online services in high efficiency and minimal cost which provide a high economic value. It is undoubtedly the next revolution in the Internet world as well as the business world.

Currently, more E-commerce enterprises move to Cloud Computing to achieve high practical value. This paper introduces an overview for Cloud computing in E-commerce through discussing various definitions for both concepts, highlighting the benefits and challenges for applying Cloud Computing in E-commerce, and discussing a suggested cloud computing E-commerce framework.

Keywords: cloud computing, e-commerce, ICT, Internet, SMEs

1. Introduction

There is no doubt that we are living in an era where things are getting old while they are still in the top of their modernity, the pace of technological development is accelerating, and hardly a day goes by without a witness appeared on the essential changes in all sectors, including the business sector.

In the past, to sell products you have to rent physically an office space which added different expenses, then E-commerce appeared and gave the flexibility for enterprises to sell products online without any need to rent a shop like before. These days, many more E-commerce enterprises especially SMEs (Small and Medium sized Enterprises) take advantage of the benefits of cloud computing (Mann et.al., 2008), where the growing of this innovation led them to compete with the large enterprises in providing products and services as they have a large infrastructure despite their limited infrastructure (Abdulkader and Abualkishik, 2013).

The benefits of investment in cloud computing technology in businesses have been widely recognized (Armbrust et al., 2010) such as flexibility, reliability, enhancing the availability, and reducing the cost of E-businesses. (Tuncay, 2010).

2. What is E-commerce?

E-commerce came into being since late 1970s. Many advantages of online shops encourage consumers to adopt, like lower costs, better prices than traditional retailers, and ability of consumers to compare prices from different retailers (Chaparro-Peláez et al., 2016, Chang et al., 2010).

The Electronic Commerce Association introduced a general definition of E-commerce: "electronic commerce covers any form of business or administrative transaction or information exchange that is executed using any ICT (Information and Communications Technology)".

Raymond (2001) defined E-commerce as "The functions of information are exchange and commercial transaction support that operate on telecommunications networks linking business partners (typically customers and suppliers)".

Turban et al. (2002) defined it as "An emerging concept that describes the process of buying, selling, or exchanging services and information via computer networks".

In general E-commerce can be categorized as:

1. Consumer to Consumer E-commerce (C2C E-commerce): The E-transactions between consumers themselves
2. Business to Consumer E-commerce (B2C E-commerce): Enterprises can sell to the consumers directly.

3. Business to Business E-commerce (B2B E-commerce): The E-transactions between Enterprises.

4. Consumer to Business E-commerce (C2B E-commerce): Consumers can sell products to the Enterprises.

Actually, there are many different types of E-commerce applications, which grouped in Figure 1 according to categories (Arie et al., 1995, Block et al., 1996, Soh et al., 1997, Zwass, 1998, Turban et al., 2000, Ainin, 2000, Fatimah et al., 2000, Fahri and Omar, 2001, Oakes, 2002, Johnson, 2003, Ainin, and Jaffar, 2003, Smith and Chaffey, 2005, Lawal, 2010).

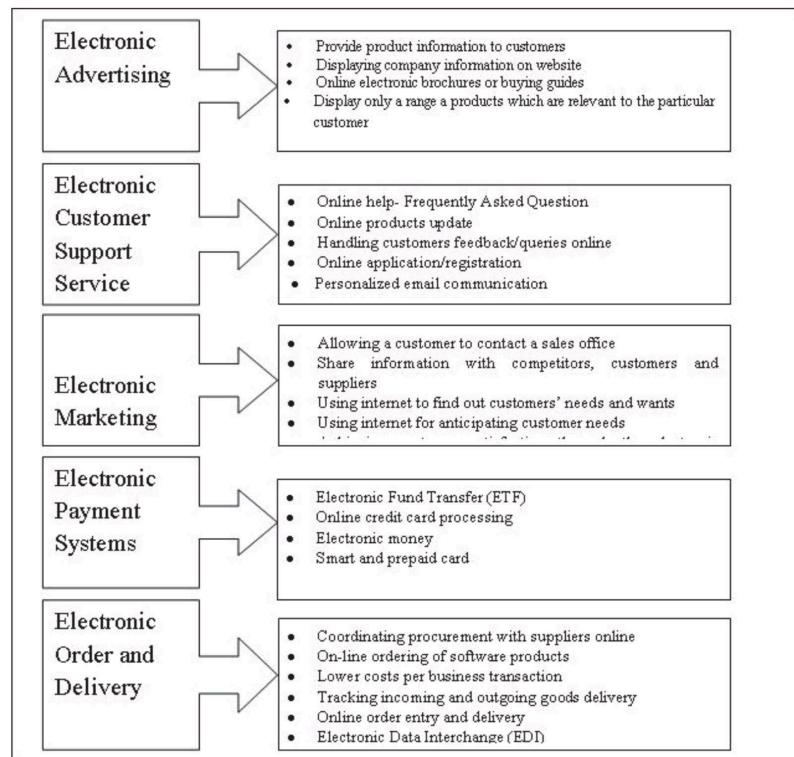


Figure 1. Applications of E-commerce

3. What is Cloud Computing?

Many governments and businesses considered it as a revolutionary term. The term of cloud computing doesn't have a unified definition at present. IEEE Computer Society defined it as (Kho, 2009): "A paradigm in which information is constantly stored in servers on the Internet and cached temporarily on clients that include desktops, entertainment centers, computers, notebooks, handhelds, etc". It is an IT tool used to deliver computing as a service not a product.

Many scientists of NIST (National Institute of Standards and Technology) defined it as follows: "Cloud computing is a model for enabling convenient access to networks and applications quickly, common set of configurable computing resources (e.g., networks, servers, storage and applications) that can work with little or interfere with the service provider to provide or be released immediately."

Other researchers like (Paul et.al, 2010) defined it as "A style of computing where massively scalable information technology related capabilities are provided as a service across the internet to multiple external customers".

As shown in Figure 2(Arron, 2017), the cloud computing has three widely referenced service models. SaaS (Software as Service): It means that the end user will deal with the site remotely over the Internet. CRM (Customer Relationship Management), and the data center which displayed by Amazon Web Services are examples for this model. PaaS (Platform as a Service): The best example of PaaS is the Google App Store. It is primarily aimed at the developers' activities who want to deploy their applications directly in the cloud server, and they haven't an interest in the connection to the infrastructure of the servers. And IaaS (Infrastructure as a Service): It

gives developers the permission to take the highest level of direct interaction with the infrastructure of servers. It also allows them to deploy their own applications in remote environment and to control them remotely. So far, the SaaS model is the dominant model in the current industry.

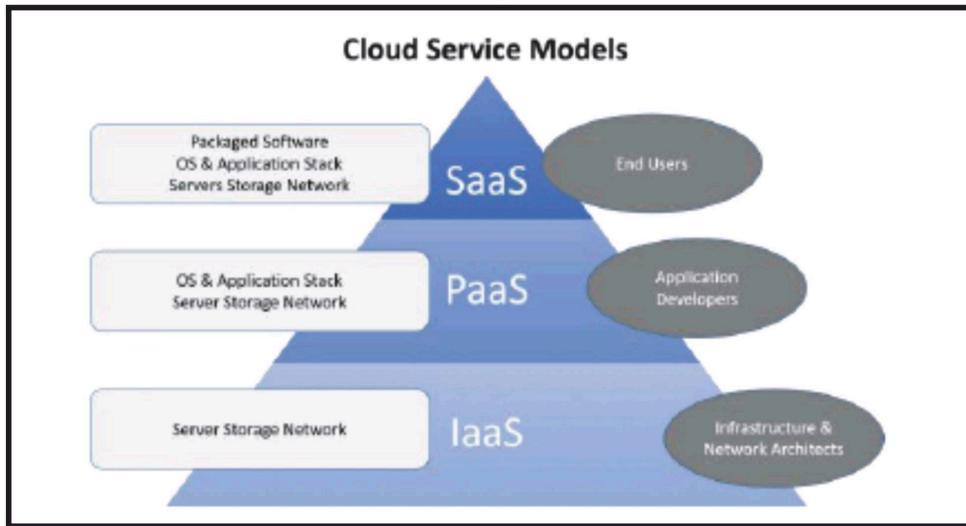


Figure 2. Models of Cloud Computing

4. Why is Cloud Computing in E-commerce?

Cloud Computing and E-commerce are now two famous terms in our daily lives. The main reason of this reputation returns to cost beneficial where cloud computing service saves enterprise's the cost of IT infrastructure, on the other hand E-commerce provides traders to do business without expenses like renting or buying a business shop.

The adoption of cloud computing and E-commerce in developing countries have been widely discussed by several researches, which showed that these innovations leveraged the developing countries in transforming into digital economy leading to global market penetration and national economic growth (Busalim and Hussin 2015; Khan et al. 2017; Nawaz et al. 2016; Rukhsara et al. 2016; Yu and Ni 2013; Dai et al. 2014; Guo and Gao 2015; Liu and He 2017; Rao et al. 2013; Neumann et.al. 2010; Talib and Alomary 2016; Goel and Goel 2017).

No one can deny that cloud gives positive opportunities and benefits for E-commerce, but smart organization should have a trade-off between costs before using it. The cloud computing allows organizations to perform business without having to develop and retain IT infrastructure. E-commerce provides the flexibility for business to sell products online without having to physically rent an office, but still there are expenses related to hardware and software resources. These days, many more E-commerce enterprises obtain advantages of the profit of cloud computing (Nevin, 2015; Jignesh 2014; Abdulkader and Abualkishik, 2013) as shown in Table 1.

Table 1. Advantages of Cloud Computing in E-commerce

| Advantage | Description |
|---------------------------|---|
| Cost saving | Reducing IT resources, installation, and implementation. |
| Scalability | The business requirements are changing constantly. Cloud computing enable rapid adaptations of IT to these changes. |
| Efficiency | IT organizations can concentrate on its businesses and get benefits through development and innovative research |
| Availability and Mobility | Through smartphones, customers can access services and products anytime and anywhere. |
| Easy management | Maintenance of hardware, software, and even infrastructure is simplified. |

According to Gartner Group (Gartner, 2016), as shown in Table 3. The worldwide public cloud services market is projected to grow 21.4% in 2018 to total \$186.4 billion, up from \$153.5 billion in 2017. The fastest-growing

segment of the market is cloud system infrastructure services (IaaS), which is forecast to grow 35.9% in 2018 to reach \$40.8 billion (see Table 2). SaaS remains the largest segment of the cloud market, with revenue expected to grow 22.2% to reach \$73.6 billion in 2018.

Gartner expects the growth rates of public cloud to stabilize from 2018 onward even though the revenue of it is growing more strongly than forecast, reflecting the status and maturity that public cloud services.

Table 2. Worldwide Public Cloud Service Revenue Forecast (Billions of U.S. Dollars)

| | 2017 | 2018 | 2019 | 2020 | 2021 |
|--|--------------|--------------|--------------|--------------|--------------|
| Cloud Business Process Services (BPaaS) | 42.6 | 46.4 | 50.1 | 54.1 | 58.4 |
| Cloud Application Infrastructure Services (PaaS) | 11.9 | 15.0 | 18.6 | 22.7 | 27.3 |
| Cloud Application Services (SaaS) | 60.2 | 73.6 | 87.2 | 101.9 | 117.1 |
| Cloud Management and Security Services | 8.7 | 10.5 | 12.3 | 14.1 | 16.1 |
| Cloud System Infrastructure Services (IaaS) | 30.0 | 40.8 | 52.9 | 67.4 | 83.5 |
| Total Market | 153.5 | 186.4 | 221.1 | 260.2 | 302.5 |

The most challenges to be faced in applying cloud computing in E-commerce (Abdulkader and Abualkishik, 2013, Nevin, 2015, Al-Jaberi et.al 2015, Babar and Chauhan, 2011, Buyya et.al, 2008) are presented in Table 4.

Table 3. Main challenges of applying cloud computing in E-commerce

| Challenge | Description |
|--------------------------|--|
| Security | It is the main challenge, where data can be accessed, modified, or even destroyed during processing or transmission. Until now, it is hard to protect programs and data and there are no effective solutions. |
| Data Privacy | It is an important challenge, until now no technical solutions to protect the clients' information. |
| Data Storage | The clients of cloud services worry regarding their inability to control of stored data place. |
| Trust | As a definition trust is “the degree by which a target object such as software, a device, a server, or any data they deliver is considered secure.” Until now, it is difficult for consumers to distinguish between good and bad E-commerce sites. This situation does not encourage enterprises and clients to move to the cloud. |
| Connectivity | In the cloud, to access shared information or resources the user must be connected to Internet. |
| Service standards issues | No available information for enterprises regarding mode of operations, technology used, and staff situation which let the clients are worrying to use cloud computing without knowing these details. |

5. How is Cloud computing work in E-commerce?

The employment of cloud computing into E-commerce will bring a tremendous business change to enterprises. Figure 3 summarizes various influences of cloud computing on SME's E-commerce (Yu J and Ni J, 2013). The impacts are obvious and multifaceted in terms of deployed technology, service, and business model.