# EXPERIMENT 05

CLASS: BE CMPN A 2 ROLL NO. : 18

NAME: REBECCA DIAS PID: 182027

# Aim:

Study Platform as a Service.

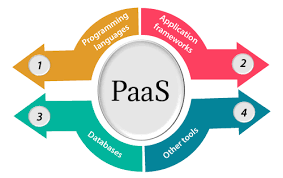
# Theory:

## Prepare a detailed study of Platform as a Service

### What is PaaS?

Platform as a service (PaaS) is an enabler for software development where a third-party service provider delivers a platform to customers so they can develop, run, and manage software applications without the need to build and maintain the underlying infrastructure themselves.

Most platforms as a service include templates or build packs, which provide an opinion as to how certain types of applications should be built, typically around the popular 12-factor methodology. This is why PaaS options are often labelled “opinionated” and are best suited for new, greenfield applications.



As with other cloud services such as infrastructure as a service (IaaS) and software as a service (SaaS), a PaaS is typically accessed over the internet but can also be deployed on-premises or in a hybrid mode. Regardless, the underlying infrastructure an application runs on is managed by the service provider. In many cases, the customer can decide where their application is physically hosted and is given a choice over how performant or secure that environment is, often at an additional cost.

### How to use PaaS (Customer)

PaaS provides direct support for business agility by enabling rapid development with faster and more frequent delivery of functionality. It does this through continuous integration techniques and automatic application deployment. PaaS also enables developers to realize the cloud’s broader benefits.

Because PaaS delivers all standard development tools through the GUI online interface, developers can log in from anywhere to collaborate on projects, test new applications, or roll out completed products. Applications are designed and developed right in the PaaS using middleware. With streamlined workflows, multiple development and operations teams can work on the same project simultaneously.

PaaS providers manage the bulk of your cloud computing services, such as servers, runtime and virtualization. As a PaaS customer, your company maintains management of applications and data.

### How to provide PaaS (Cloud Service Provider)

When organizations adopt a SaaS solution, they are choosing to outsource their entire technology stack and the associated maintenance costs to a third-party provider. PaaS, in contrast, does not typically replace an organization's entire IT infrastructure, rather, it helps organizations access key services with minimal start-up costs and reduced time to deployment.

PaaS products are generally geared towards software development, providing users with resources such as computing power and data storage capacity on demand, along with functions like version management, text editing, testing services and more. A PaaS product can also facilitate collaboration between geographically distant team members, allowing them to access the same development environment from separate locations.



PaaS providers typically price their services using a pay-per-use model. Depending on the vendor, usage can be calculated in different ways. One vendor might charge a fixed rate per user based on a limited number of custom app objects, so a single user with 10 objects might cost $x per month while the same user with 2000 objects costs $xxx per month. Another vendor might charge based on the number and speed of servers and the overall bandwidth used. The usage of computing instances, the volume of data storage required on the platform and the amount of outbound traffic are all typical factors when determining the price of a PaaS subscription.

## Advantages and Limitation of PaaS

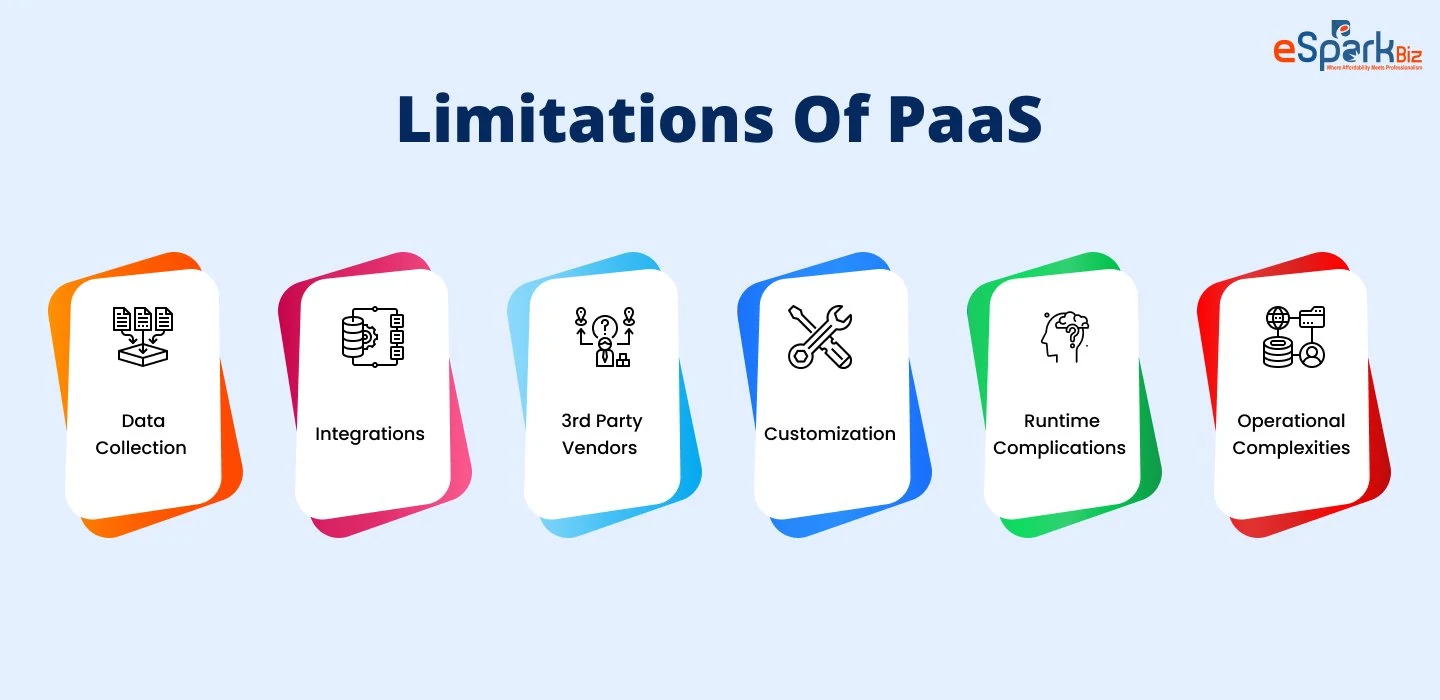
*Advantages of PaaS*

By delivering infrastructure as a service, PaaS offers the same advantages as IaaS. But its additional features—middleware, development tools and other business tools—give you more advantages:

* Cut coding time. PaaS development tools can cut the time it takes to code new apps with pre-coded application components built into the platform, such as workflow, directory services, security features, search and so on.
* Add development capabilities without adding staff. Platform as a Service components can give your development team new capabilities without your needing to add staff having the required skills.
* Develop for multiple platforms—including mobile—more easily. Some service providers give you development options for multiple platforms, such as computers, mobile devices and browsers making cross-platform apps quicker and easier to develop.
* Use sophisticated tools affordably. A pay-as-you-go model makes it possible for individuals or organisations to use sophisticated development software and business intelligence and analytics tools that they could not afford to purchase outright.
* Support geographically distributed development teams. Because the development environment is accessed over the Internet, development teams can work together on projects even when team members are in remote locations.
* Efficiently manage the application lifecycle. PaaS provides all of the capabilities that you need to support the complete web application lifecycle: building, testing, deploying, managing and updating within the same integrated environment.

*Disadvantages of Paas*

* Security:All the data of applications are stored inside the provider's cloud database. This brings up confidential issues since the private and sensitive information can be viewed by the provider. As a result, it is the responsibility of the businesses to secure their applications by selecting a trustworthy provider. Otherwise, the existing information could be at risk.
* Control:Users lack some controls over a PaaS solution. It is dependent on the providers capabilities. For an example, whenever the provider increases their pricing scheme similarly the applications could become expensive too. Therefore, it is important to choose your PaaS provider wisely.
* Reliability: PaaS solutions often face reliability concerns. When using PaaS, users must be ready to face frequent downtimes. There can be power outages, disasters or other misfortunes all which can lead to devastating consequences. This can impact both business operations and customers who use the application. As a result, the users must perform their own backups to avoid data losses.
* Compatibility: Not all the components are cloud enabled. Whatever the products offered by the PaaS provider is customized themselves. Due to this the infrastructures may face compatibility issues in a new environment.
* Integration: Similar to compatibility, each and every PaaS provider has their own integration process. Two different PaaS solutions cannot be integrated with each other. If tried contradictions can be seen. Moreover, once there is integration issues it is not an easy task to change the PaaS provider.
* Locked-in Features- Some of the features offered by the PaaS may be completely irrelevant to the user. It can be a program, language or an interface that is locked-in. The users have no authority to change it. The only thing the users could do is either make changes to the application or rebuild it.



## Study security issues in PaaS

PaaS allows companies to build, run and ultimately manage Web applications without the infrastructure that is normally required.

Since PaaS is based on the notion of using shared resources (such as hardware, network, and security provisions), security concerns are usually focused on mission-critical information that hackers can obtain during a data breach. If the PaaS tenants have Administrator/’root’, or shell access to the servers running their instances, additional security issues could arise if hackers are able to gain unauthorized access and change configurations. Additionally, security controls and self-service entitlements offered by the [PaaS platform could pose a problem if not properly configured](https://docs.microsoft.com/en-us/azure/security/fundamentals/paas-deployments). Providers should be able to provide clear policies, guidelines, and adhere to industry-accepted best practices.

Once again, security cannot be solely the PaaS provider responsibility. When selecting a PaaS vendor, consider these crucial issues before final selection:

* What are the types of encryption used?
* What are the data independence and availability? (Can you move your virtual machines and all of their data to another provider? Who has access to it? What happens if a cloud instance migrates to another country?)
* What are the disaster recovery/business continuity protocols?

Security issues are generally centred on mission-critical information that hackers may access during a data breach since PaaS is built on the concept of sharing resources (such as hardware, network, and security provisions).

Additional security concerns may occur if hackers are able to obtain unauthorized access and alter configurations if PaaS tenants have Administrator/’root’, or shell access to the servers running their instances.

In addition, if the PaaS platform’s security controls and self-service entitlements aren’t properly configured, they could cause problems. Providers should be able to have consistent policies and procedures, as well as follow industry standards.

Once again, responsibility for security cannot be exclusively the of the PaaS provider. Before making a final decision on a PaaS provider, keep the following points in mind:

• What is the data center availability and independence? (Are you able to transfer all of your virtual machines and their sensitive data to a different provider? What kind of people have access to it? What if a cloud instance moves to a different country?)

## Technologies used to provide PaaS

### [Amazon Elastic Beanstalk](https://aws.amazon.com/elasticbeanstalk/)

[@awscloud](https://twitter.com/awscloud)



Amazon Elastic Beanstalk gives users an easy way to deploy and provision cloud-based resources that automate setting up applications on Amazon Web Services. All you need is to upload your applications, and everything from load balancing, provisioning, application health monitoring, and auto-scaling will be handled by Amazon Elastic BeanStalk.

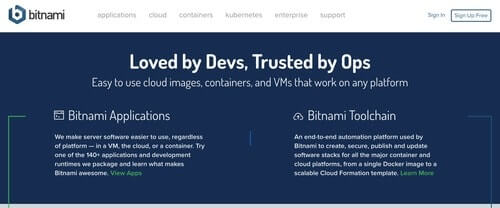
Key Features:

* Support for Java, PHP, Python, .NET, Ruby, and Node.js applications, as well as other Web development stacks and programming languages
* Support for applications that are not written to be used on the Web
* Complete control of the resources that your applications will be using
* Automatic scaling of your application based on the resources needed

Cost: No additional fees, but you pay for the AWS services and resources that you use to run your applications.

### [BitNami](https://bitnami.com/)

[@bitnami](https://twitter.com/bitnami)



Bitnami enables you to host and run your applications in the cloud. You can have Bitnami stacks deployed and managed easily and quickly. It supports different cloud platforms including Amazon Web Services, Google Cloud Platform, Microsoft Azure, and Oracle Cloud Platform. Apart from cloud services, Bitnami also gives you applications that help deliver the same experience for different users no matter what platform they use: containers, on the cloud, or with a virtual machine.

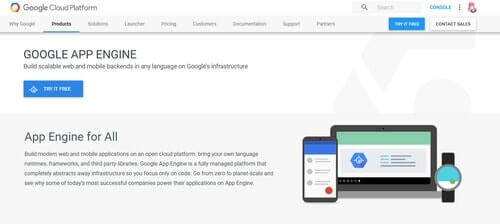
Key Features:

* Supports applications built for or with WordPress, Drupal, Redmine, Moodle, GitLab Community Edition, LAMP, Mean, Redis, PostgreSQL, and other stacks
* Very secure with the ability to plug vulnerabilities as soon as they are discovered
* All Bitnami applications are self-contained and include all the required runtimes, databases, and libraries

Cost: Pricing for Bitnami Applications varies, with free trials available for some applications. Cloud hosting price is unavailable.

### [Google App Engine](https://cloud.google.com/appengine/)

[@googlecloud](https://twitter.com/googlecloud)



Google App Engine helps you build mobile and Web back-ends using the programming language, frameworks, libraries, and runtimes that you are most comfortable with. Google takes care of the infrastructure while you work on your code. Check out our handy [feature comparison chart](https://stackify.com/microsoft-azure-vs-amazon-web-services-vs-google-compute-comparison/) to compare Google Compute with Azure and AWS.

Key Features:

* Google App Engine natively supports Java, Node.js, C#, Ruby, Python, PHP, and Go
* Open source to help you avoid lock-ins
* Supports SQL and NoSQL databases
* Offers a wide variety of monitoring, troubleshooting, and other helpful developer tools
* Can host different application versions
* You can split traffic to different versions

Cost: Free trial is available. Billing depends on standard or flexible environments. For standard environments, pricing starts at $0.05 per instance per hour. On the other hand, flexible environment instances start at $0.0526 per core hour, $0.0071 per GB hour, and $0.0400 per GB per month.

# Activity:

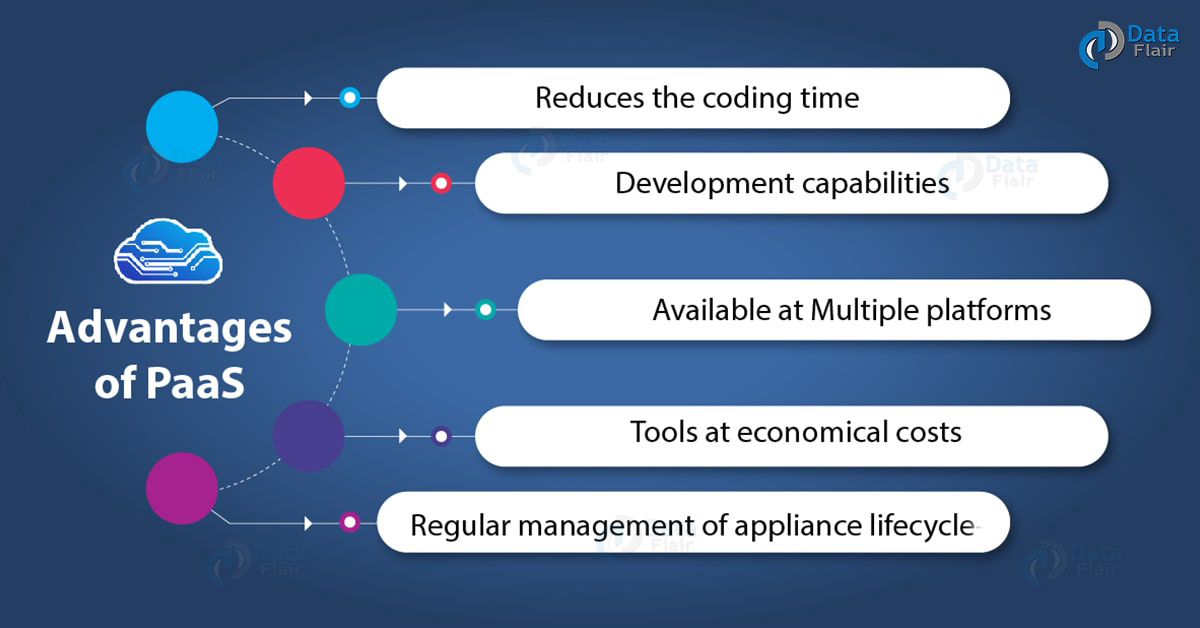
## Use any suitable cloud service, providing a platform as a service

## Deploy an application

## Access it from a remote machine

# Conclusion:

## What are the benefits of using Paas



1. Cost

All the companies which uses PaaS enjoy low investment cost since it does not require hardware and software. There is no expense involved in setting up and maintaining the software. The software can simply be used without needing to invest in infrastructure.

Apart from installation cost, even the expense of developing and testing the apps are significantly reduced. Therefore, it is preferred most by companies with limited resources willing to lower their operating cost.

2. Availability

PaaS is made available to all the professionals irrespective of the location. As a result, the employees of an organization can communicate with each others using a single environment. For an example, different staffs involved in testing and maintenance can collaborate with each other to carry out their works from different locations. Thus, the PaaS is able to increase the employee productivity.

3. Scalability

The scaling mechanism behind PaaS makes it a reliable scalability solution. The environment used in a PaaS is highly scalable with resources and tools. The structure is flexible enough to allow business to expand without high investment. This is especially beneficial for small businesses looking to increase their resources and expand their service.

4. Productivity

The work environment provided by the PaaS is already existing. They offer a pre-defined business plan. Meaning, each project needs not to be implemented from the beginning. Thus, the users does not create their own to develop new applications. Users can follow a standardized technique for the application development.

5. Upgradeability

Updates are really necessary for PaaS solutions to keep up with the competition. If the updates are not given time to time, the users could constantly miss out new features and improvements. These updates always ensure that the applications are running in their latest versions. Now the updates are automatically delivered by the PaaS provider.

6. Platform Support

It is crucial for a business to develop apps that support wide range of devices and operating systems. PaaS developers always make sure that their apps are supported by different platform types. For this they use specific tools designed for this purpose. Thus, PaaS ensures compatibility.

# References:

1. [What is PaaS? Platform-as-a-Service Explained | Sumo Logic](https://www.sumologic.com/glossary/paas/)
2. [Practical guide to PaaS: Benefits and characteristics](https://www.ibm.com/blogs/cloud-computing/2016/08/22/paas-benefits-characteristics/)
3. [What is PaaS? A simpler way to build software applications | InfoWorld](https://www.infoworld.com/article/3223434/what-is-paas-a-simpler-way-to-build-software-applications.html)
4. [6 Advantages and Disadvantages of PaaS | Drawbacks & Benefits of PaaS](https://www.hitechwhizz.com/2021/06/6-advantages-and-disadvantages-drawbacks-benefits-of-paas.html)
5. <https://www.arrow.com/globalecs/na/arrow-channel-advisor/saas-paas-and-iaas-what-are-all-the-risks>