Journal of Engineering Excellence
ISSN: 3108-2793 | Volume 1 Issue I October-December 2025 | Inaugural Edition
www.jeejournal.com JEE-01

AGILITY IN THE CLOUD: A HYBRID MODEL FOR SCALABLE AND CONTINUOUS DEPLOYMENT

Dr. B. RAMESH, *Associate Professor, Dept of CSE,*
SREE CHAITHANYA COLLEGE OF ENGINEERG, KARIMANGAR, TG.

ABSTRACT: Software development underwent radical transformation in the mid-1990s, when the concept of rapid development gained traction. Depending on the needs of the client, brief iterative cycles of output are prioritized. Quick feedback and agile development were still not supported by any current technologies. Because it eliminated the need for widely dispersed demands, which can impede progress, cloud computing was the connecting piece. The primary objective of this research is to examine the relationship between cloud computing and agile software development. Software development is both practical and innovative when we look into this connection. Building a mobile app for warehouse management on Google Cloud Platform utilizing the fast prototyping process of the Dynamic Systems Development process (DSDM) exemplifies the overall points made in this article. Warehouse management doesn't typically employ agile software development methodologies.

Keywords: Agile development, cloud computing, DevOps, continuous integration, continuous delivery, Scrum, Kanban, automation.

1. INTRODUCTION

Agile software development allows businesses to receive the software they require more quickly and with less effort. In an agile environment, everyone's demands are prioritized by involving them in the process. Take advantage of the tools and resources that are constantly evolving to meet the needs of creators and consumers. Developers have the ability to fulfill their clients' desires. The final round involves letting users try out the updated features and provide comments. By carefully listening to customer feedback, system makers can improve the way things work. Producers and consumers alike can benefit from the streamlined participation offered by cooperatives. Service quality, cost, and waste can all be improved with the use of cloud computing and agile methodologies.

It is critical for cloud services to promptly address client demands. When working on a large project, it can be helpful to employ agile techniques to divide the work into smaller, more manageable chunks. Clients are surveyed using this method following the completion of each project. Each component is planned, constructed, and tested to ensure it meets the requirement and minimizes difficulties Integrating cloud applications with agile methodologies has numerous advantages.

Some of the advantages include lower costs, quicker product releases, more efficient use of resources, and better application standards.

This research provides strong support for the premise that combining agile practices with cloud computing yields positive results. The advantages of using cloud computing services and agile

methodologies are obvious. The practical applications of these findings will be explored in the subsequent sections of the research. Several cloud models were discovered after investigating the selected organization's structure and operations. They are as follows. Next, we'll take a closer look at agile development and its potential applications in warehouse management. Here you will find a rundown of the software's key features. Considering the benefits and drawbacks of each, we compare traditional agile development with cloud-based agile development. The two expansion kinds are functionally equivalent. The first one employs agile techniques, while the second one makes use of cloud-based solutions. The conclusion is supported by this disparity.

2. SPRINT CLOUD COMPUTING

This discussion will center on Sprint's cloud computing and phone services.

Sprint company - introduction

When it comes to software development and other digital needs, Sprint works with companies of all shapes and sizes. Plus, they're pros when it comes to creating and utilizing software. The company's primary areas of operation include financial management, software design, and information technology. When starting a firm, it's important to have tools that can assist with strategy development, resource allocation, and task tracking.

Gaining a competitive edge, improving IT efficiency, lowering running costs, and making operations more flexible were the reasons Sprint chose cloud computing over traditional computer architecture. The efficiency of Sprint's integration with the cloud was tested using a mathematical model. This one-of-a-kind occurrence piqued people's curiosity in cloud computing. For anyone curious about cloud computing, this mathematical framework is an excellent starting point.

Companies should consider a lot of factors while deciding on the right cloud model. Factors influencing output include project scope, infrastructure, security standards, and organizational objectives. After extensive research, the organization settled on a cloud-based solution. The next portion goes into more depth on this topic.

Sprint adopted cloud models

Figure 1 depicts the Sprint hybrid assembly. Businesses adopt hybrid cloud solutions, which combine public and private cloud computing. Access to private cloud networks, devices, data, and applications is available to the public. The staff working on the program might make use of Google's public cloud service. Google App Engine provides a wealth of resources for app creation and testing. After analyzing the company's operations and infrastructure thoroughly, Sprint determined that a hybrid deployment method would be the most suitable for the business.

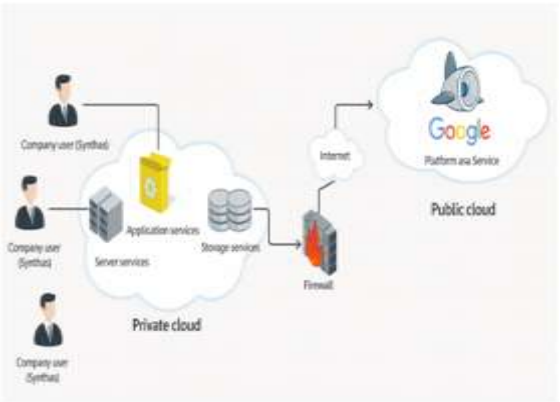

Figure1. The spring hybrid implementation approach combines methodologies for a more complete and effective solution.

Sprint must weigh the benefits and drawbacks of public and private cloud options in order to competently manage a hybrid architecture. The following private cloud capabilities were deemed necessary by Sprint. The firm serves many different types of businesses by providing them with information technology services. Utilizing company data is acceptable. It is necessary to implement measures to ensure the security of information in order to keep control. There is a severe lack of utilization at the company's data center. Even with additional software, the expenses of utilizing the data center for cloud storage would still be higher than those of the public cloud.

In order to ensure compatibility during sprints, where numerous individuals utilize software to collaborate on projects, public cloud services from various providers were utilized. Assess the quality of the application's code by observing its writing and testing.

After software initiatives, such as PAAS, are completed, the business provides software and hardware SaaS solutions. These can be group projects or individual consultations. Google Cloud is where you can obtain this tool. Software as a service is what it stands for. With Google App Engine, developing apps is a breeze. After this, we will go on to discussing the SaaS WMS.

3. CASE STUDY

In this section, we will examine two approaches to application-based development. In what follows, we'll compare the services provided before and after Sprint used cloud computing to build the app.

Commonalities of the development of theapplication

Rapid WMS application development using DSDM was proven to be the optimum solution in every situation. The DSDM scheme is composed of three primary elements. As part of the pre-project phase, you should establish your project objectives and prioritize the feasibility studies you will do. We have done our best to incorporate all the essential professional services into the next section of the study. Part two of the project included the distribution of many primary and secondary assignments.

When the operational system is changed, it is done so with a practical and economic perspective in mind. Iterative design and construction involves repeatedly conducting research, designing, implementing, and evaluating in order to improve a system or product. Because of this approach, things can continuously improve and evolve in response to user feedback and actions. During synthesis, a vast number of components are assembled and mixed to form a functional whole.

The final stage, commonly referred to as the "post-project phase," involves evaluating the project's components to ensure they are functioning optimally. Maintenance procedures that comply with DSDM are used for this. The lack of professionalism makes this section incomplete. This Advanced Software Engineering course was attended by four students from Sarajevo's Electrical Engineering Faculty. Concepts, programming, and software testing were all collaborative efforts by all team

members. Additionally, DSDM assignments have been distributed. The tool was heavily utilized by both novices and seasoned professionals in the field.

Sprint planning is covered in this section. The selected firm will oversee the preliminary phases of the project. Each group was assigned a different organization to develop an app for in order to demonstrate the usefulness of cloud computing and agile approaches. Get your app redevelopment plans in motion and prepare to transfer to the cloud.

It always took three games to attain the goal. The amount of time invested in researching the subject was proportional to the amount of revisions made throughout the production process.

The program monitors the building's activities. Additionally, it provides customers with comprehensive reports and additional information to aid in comprehending the warehouse's operation.

Just to illustrate the point. Connected program sections and module input/output are shown in Figure 2.

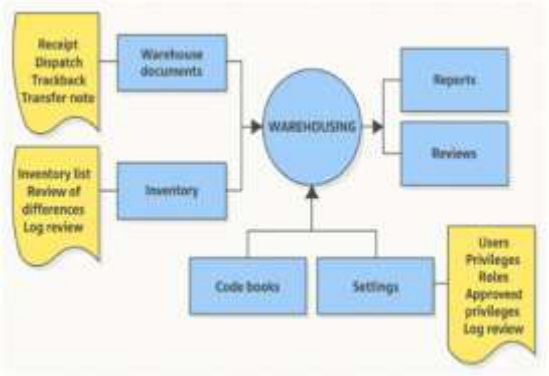

Figure2. Functionality of logistics management software

Application development without use ofcloud computing

A web app built with the ASP.NET framework, Visual Studio, and Microsoft SQL Server served as the product's foundation prior to its migration to the cloud. Collaborating as a team and repeatedly adjusting our strategy yielded the best outcomes.

It is imperative that all team members have SQL Server and Visual Studio installed on their own computers with the most recent versions.

Each iteration ended with students turning in a CD containing the software's working code to their lecturer for grading. It was necessary to make multiple copies of the merged program in order to test all three ways.

Cloud based application development

The game was created using a cloud-based application known as Google Game Engine.

Game developers utilize web-based tools. Novel concepts emerged during a total solar eclipse. Google App Engine app development is made feasible with Eclipse. Devices hosted in the cloud were utilized to execute tests concurrently throughout the testing procedure. Following development, the initial app version was uploaded to the Google App Engine Platform for educators to evaluate. What had changed and what was required today was obvious. The instructor now has the option to teach the class online. Prior to their assembly, the final components of rounds two and three were modified. The most recent application was forwarded to the instructor for review.

The scenario's concepts and procedures remained unchanged despite the fact that the development platforms changed. Everyone in the team that worked on the app, from brainstorming to creating the

final version, pitched in.

4. AGILE METHODS AND CLOUD COMPUTING

An integral aspect of agile development is iterative and gradual development. The typical duration of an iteration in software development is two to four weeks. With each cycle, there is an improvement. While agile development is a continuing activity, cloud computing refers to the continuous supply of software to end users. Rapid resource deployment is possible with cloud applications for agile project teams. Agile project teams continue to reap benefits for cloud service providers. The importance of collaborating and communicating across process and technology is emphasized in the Agile Manifesto. Music that introduces or concludes a speech is thus considered an obnoxious form of speech. With cloud computing, less effort and time is required to maintain devices operational. Groups developing cloud-based items may have an easier time getting their products into production. By eliminating tedious and time-consuming tasks, cloud solutions make software installation easier for IT workers. The conventional on-premises method involves a group of administrators installing the software and configuring and updating it locally. All users collaborate on this process. Development becomes quicker, cheaper, and better able to meet needs when cloud computing and agile methodologies are combined.

Gains during agile development with cloudcomputing

Cloud computing is now integrated into the iterative development process on multiple levels. The aforementioned alterations improve the speed and quality of application development throughout multiple phases of agile development.

The findings can be seen at this location:

- One definition of a building is an architectural structure.
- Interactions between people are frequent.
- The ability to exchange goods and services is made possible by models.
- Sending in your documents and applying

Making these adjustments could make the user's text sound more intelligent. Academic research relies heavily on reevaluating and integrating data. By adding. Versions and prototypes do exist, no doubt about it. Making software requires a strategy that can be updated easily, thus building infrastructure comes first. Agile approaches can get pricey when cloud software isn't an option. Hardware, software licensing and updates, network infrastructure modifications, testing and tracking tools, security product setup and maintenance, and other related expenses all fall under this category. Combining cloud computing with quick operations reduces threats to the IT infrastructure. Cloud computing's adaptability means it can work with a wide variety of software and hardware platforms. The expense of creating an app can be reduced by doing away with the requirement for costly software and tools. This technology distributes resources dynamically according to user demand.

Teams who have trouble communicating often end up with abandoned initiatives. Cloud computing is the answer to this

dilemma. Sharing information and resources is made easier with many Software-as-a-Service (SaaS) solutions. When a project is still in its early phases, team members may have divergent priorities and interests, making it difficult to communicate effectively. The use of cloud storage for group model sharing, review, and updating significantly shortens each cycle.

Cloud computing is a great tool for managing an application's scope. Due to their ease of use in developing cloud-based applications, these technologies are well-suited for usage in the cloud. In most cases, customers will do testing at the development team's office. A bottleneck occurs when there are insufficient instances of the operating program to handle the site load. Thanks to cloud computing, the software is accessible from any location so long as there is a reliable Internet connection. More cloud computing testing can be supported by gradually adding to the present codebase on the cloud. The cloud-based testing tool can be used by users or testers with just the testing environment's URL. Readers can review the modification and provide feedback.

The need to mislead and lie decreases as feedback loops lengthen. The level of client satisfaction directly correlates to the amount of time and effort required to develop software. When users use apps hosted in the cloud, providing support becomes a breeze. Users may simply think of ideas, describe their wants, and determine if cloud computing is a suitable fit because it is so accessible. Local employees can use a cloud-based application to create and distribute prototypes, which clients can request in order to obtain feedback more quickly. The use of unapproved prototypes eliminates risks associated with on-premise installations, which necessitate substantial investments in infrastructure.

Comparing the development of application when using agile methods with and withoutcloud computing

Figure 3 displays the agile development technique and all of the project's iterations. The duration of each stage of agile development is illustrated in Figure 3. You can see the length of the steps in Figure 1. The aforementioned advantages have a significant impact on 1.3.

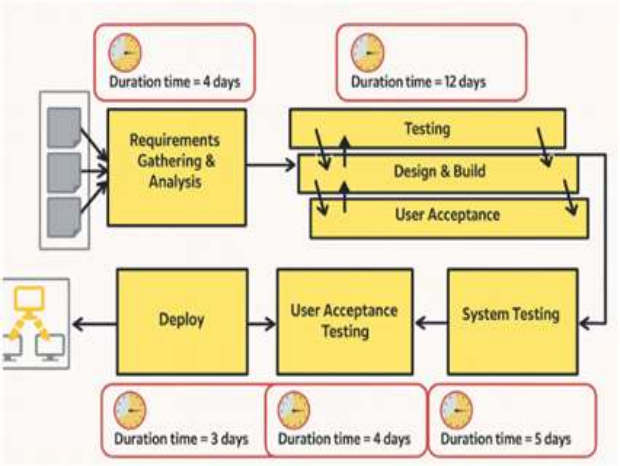

Figure 3: Cloud and agile development have been proved to reduce development times by days.

Figure (Diagram 1). Table 4 shows that cloud computing and agile app development take varying amounts of time.

Journal of Engineering Excellence
ISSN: 3108-2793 | Volume 1 Issue I October-December 2025 | Inaugural Edition
www.jeejournal.com JEE-01

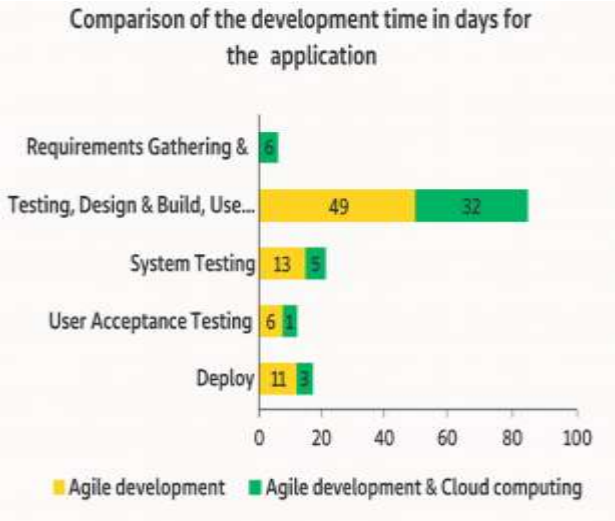

Figure 4: Combining cloud computing and agile methods drastically saves development time, improving efficiency.

To maximize the benefits of cloud computing and agile development, more extensive measures are required. The aforementioned combination offers multiple advantages.

More people were able to attend meetings and learn how the program works even when the client isn't there thanks to cloud-based technology like Skype. Making surveys and distributing school papers is a breeze with Google Docs and Forms.

One tool that simplifies app development is the Eclipse IDE. Make use of Eclipse's more powerful features to integrate repositories more quickly or create code metrics. Forms and devices that were formerly commonplace are now mostly obsolete. Software development was challenging due to the absence of cloud computing. There was a delay in the commencement of the project since many tools couldn't communicate with one other. The project went over budget and took two weeks longer than expected.

When compared to cloud setup, on-site installation is more expensive, time-consuming, and labor-intensive. By utilizing the Google application Engine with the Google Eclipse Plug-in, you are able to verify the complete or expanded application. Users of web browsers expedited the process of program user comment testing. An additional component was required due to the incorporation of cloud computing into the project's second iteration. This necessitated the allocation of additional resources. Because cloud computing simplified resource management and change management, the phase lasted the same length of time.

5. CONCLUSION

This study investigated the various benefits of combining cloud computing with agile software development as a novel approach and method for improving processes. By analyzing its resources and current state, we may learn why each organization has opted for cloud computing. Cloud computing and the Dynamic Systems Development Method (DSDM) approach are demonstrated through a case study of a warehouse management software. There are some similarities between agile software development and cloud-based agile. The advantages of the alternative approach will be made very plain. Cloud computing is widely believed to be the industry's wave of the future. Cloud computing allows users to install software by downloading it using a web browser. Agile software development methodologies allow for frequent changes to software specifications while simultaneously producing high-quality software through the development and delivery of programs.

REFERENCES

1. P. Abrahamsson, O. Salo and J. Ronkainen, Agile software development methods: Review and analysis, 2002.
2. S.C. Misra and A.Mondal, Identification of a company's suitability for the adoption of cloud computing and modeling its corresponding Return on Investment, Mathematical and Computer Modelling, pp. 1-10, 2010.
3. E. Burke, Categorizing Data Sensitivity for Computer Security, Interface, nb. 222, 2002.
4. J. Hurwitz, R. Bloor, M. Kaufman and D. Halper, Cloud Computing For Dummies, Indianapolis: Wiley Publishing, Inc., 2010.
5. Manifesto for Agile Software Development, Agile Alliance, 2001. http://agilemanifesto.org/
6. wikidot, 20. August 2011.
7. M. Granville G., The Characteristics of Agile Software Processes, IEEE Computer Society, pp. 1-3, 2001.
8. Google Developers, Google, 4. July 2012. https://developers.google.com/appengine/