

HARNESSING AGILITY: EXPLORING AGILE METHODS FOR CLOUD COMPUTING ENVIRONMENTS

#¹GUGULOTHU SAI CHARAN,

#²BANOTHU ANIL KUMAR,

#³PEDDI KISHOR, *Associate Professor*,

Department of Computer Science and Engineering,

SREE CHAITANYA INSTITUTE OF TECHNOLOGICAL SCIENCES, KARIMNAGAR, TS.

ABSTRACT :After rapid development became popular in the mid-1990s, software development altered drastically. Short, iterative manufacturing cycles driven by client input are emphasized. Agile development, however, had no way to encourage rapid iterations. Cloud computing revealed the missing linking component without time-consuming distribution requests that could stall progress. The main objective of this research is to understand how cloud infrastructures and agile software development interact. Explaining this relationship shows software development's benefits and advances. We use the fast prototyping Dynamic Systems Development Method (DSDM) to build a mobile warehouse management app on Google App Engine to exemplify the theoretical concepts in this study. Agile software development is rarely used in warehouse management.

KEYWORDS: Agile methods, Cloud computing, Agility

1. INTRODUCTION

Agile software development addresses corporate desire for faster and more effective software development.

Agile methods seek to satisfy all stakeholders' needs through collaboration. Use the ever-changing software and features available to consumers and developers. Developers can satisfy consumers. People rate and remark on new features in the final phase. System designers can improve functionality by using user feedback. In a cooperative environment, producer and consumer can cooperate actively. Agile methods and cloud computing can boost service quality, lower costs, and reduce inefficiencies.

Services in the cloud must meet user needs quickly. Agile methodologies typically break down project requirements into smaller, more manageable pieces. Consumer feedback is collected after each project using this method. Planning, building, and testing individual parts ensures quality and avoids difficulties.

The benefits of cloud computing with agile methods are many.

- Higher application standards,
- resource efficiency,
- faster product availability,
- cheaper product prices

This study provides empirical evidence for the benefits of combining cloud computing and agile ideas. Cloud computing services and agile procedures are briefly reviewed, focusing on their benefits. The following sections of this study will discuss how these results apply to real-world scenarios. Considering the selected company's operations and architecture revealed several cloud models, which are mentioned below. The next part provides a comprehensive description of agile development and its application to warehouse management. This section describes the software's main features. After assessing the benefits, we compare cloud-based agile development to traditional agile development. Both expansion types have similarities. The former uses agile methods,

whereas the latter leverages cloud technologies. This juxtaposition supports the conclusion.

2. SPRINT CLOUD COMPUTING

Sprint's cloud computing and telecom services will be covered here.

Sprint company - introduction

Sprint provides IT and software development services to businesses of all sizes and sectors. Computer application design and implementation are also their specialties. Financial management, software development, and IT are the company's primary business areas. Software solutions for work completion, resource management, and strategic planning are needed to build a company.

Sprint chose cloud computing over traditional computer infrastructure to gain a competitive edge, improve operational agility, reduce operating costs, and boost IT efficiency. A mathematical model assessed Sprint-cloud computing compatibility. This first incident sparked interest in cloud computing. If someone is considering cloud computing, start with this mathematical model. Which cloud model is suitable for a business depends on many factors. Company objectives, security protocols, infrastructure, and breadth affect production. After extensive examination, the company picked a cloud-based strategy. The following section provides more details.

Sprint adopted cloud models

Figure 1 shows Sprint hybrid installation. Organizations use internal and public cloud services in their hybrid cloud infrastructure. Users can access private cloud networks, servers, storage, and apps. The software development team may use Google's public cloud platform. Software development and testing platform Google App Engine. After reviewing the company's infrastructure and operations, Sprint decided that a hybrid deployment approach will be most successful.

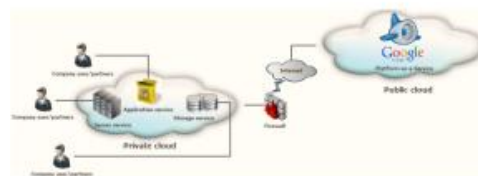


Figure 1. The spring hybrid implementation approach combines methodologies for a more complete and effective solution.

Sprint must consider the pros and cons of private and public clouds to manage a hybrid architecture. Sprint found the following private cloud characteristics essential.

Information technology services for a variety of business clients are the firm's main focus.

Using company data is appropriate. Information security is essential for control.

A large part of the company's data center is unused. Using the data center as a cloud storage facility would cost more than using the public cloud, even with additional software.

Sprint compatibility was assessed using several public cloud services:

When many people employ routine-task software.

following the app's great code development and testing.

when shared or one-time software projects like PAAS are done.

For software and hardware, the company offers SaaS options. Google cloud offers this utility for rent. Platform-as-a-service Google App Engine streamlines app development. Next, we'll cover the SaaS warehouse management program.

3. CASE STUDY

We'll compare two application-based development methods in this section. The next section will examine how Sprint used cloud computing to build the application and what services were available before and after.

Commonalities of the development of the application

All cases showed that DSDM was the best agile WMS application development method.

The DSDM model has three main parts. The pre-project stage involves setting project goals and prioritizing the feasibility research. Every

effort has been made to include all necessary expert services in the future study phase. Many primary and secondary duties were assigned in the project's second phase.

Practical and economic assessments are done for operational framework changes.

By repeating analysis, design, implementation, and evaluation, iterative design and construction improves a product or system. User input and experimentation enable continuous improvement and adjustment with this strategy. In this method, many components are assembled and integrated into a functional whole during synthesis.

In the last phase, or post-project phase, every project component is checked for optimal performance and efficiency. This is achieved using DSDM-compliant maintenance approaches. This section is unfinished since the initiatives aren't professional.

Four students took Sarajevo's Electrical Engineering Faculty's Advanced Software Engineering course. All team members participated in testing, coding, and brainstorming. DSDM assignments were also given. Subject matter expert and apprentice utilized the app frequently.

This section discusses sprint organization. The indicated firm will lead the project's initial stages. To demonstrate the huge benefits of agile methodology and cloud computing, each group had to create an application for a different business. Second, plan the cloud computing move and start application redesign. The goal was always achieved in three cycles. Iterations during development were proportional to topic research duration.

Warehouse activity is tracked by the software. It also provides thorough reports and other data to understand warehouse operations.

As an example. Figure 2 shows module input and output and software component linkages.

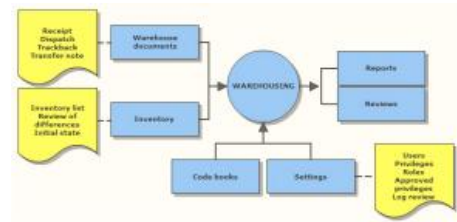


Figure 2. Functionality of logistics management software

Application development without use of cloud computing

Before moving to the cloud, this product was developed as an ASP.NET web application with Microsoft SQL Server as the database and Visual Studio as the IDE. Multiple design iterations and extensive teamwork yielded the best result.

Each team member had to install the latest SQL Server and Visual Studio versions on their computers.

After each iteration, students had to submit the software's functional code to their lecturer on a CD for review and grading. After the three steps, many integrated application copies were made for testing.

Cloud based application development

The game was made using Google Game Engine, a cloud-based platform.

Game development uses web-based methods. A total solar eclipse fostered creativity. Eclipse lets you build Google App Engine apps. During testing, cloud-based devices ran parallel tests.

The first version of the app was provided on Google app Engine Platform for instructor testing after development. This helped identify changes and new needs. The instructor can now access the course online. This was done to the final components of the second and third rounds before assembly. The instructor received the latest application for comments.

Even when development platforms evolved, scenario principles and methods remained consistent. Same team members built the app's final version and original concept.

4. AGILE METHODS AND CLOUD COMPUTING

Agile development emphasizes gradual and iterative development. Two to four weeks are normal for software development iterations. An increment is produced after each cycle. Cloud computing distributes software to end users continuously, while agile development generates products continuously. Agile project teams can deploy resources fast with cloud computing. Agile project teams constantly benefit cloud service providers.

Process and technology require collaboration and communication, according to the Agile Manifesto. Thus, starting or ending discussions with music is unlawful. Cloud computing reduces instrument maintenance time and effort. Cloud-based application development teams have simple access to production. Cloud solutions simplify software installation for IT teams by eliminating lengthy and time-consuming procedures. The usual on-premises solution requires all users to collaborate to install the software, utilize a team of administrators to install locally, and then update, which takes time. Cloud computing and agile methods have been shown to improve development efficiency, cost, and time-to-market.

Gains during agile development with cloud computing

The agile development method has used cloud computing at several levels. The above modifications increase application development quality and speed and affect numerous agile development stages.

Results are shown in:

Structures can be architectural constructions.

Interactions are frequent.

Models help trade goods and services.

Applications and submission procedures

The following adjustments could make the user's text sound more erudite: Data integration and reevaluation are essential to academic research. By incorporating

Certainly, prototypes and iterations exist.

Software development requires a flexible design, therefore infrastructure must be built first. When cloud computing is unavailable, agile methods are expensive. These costs

include software licenses and updates, hardware, network infrastructure changes, monitoring and testing tools, security product implementation, and maintenance. IT infrastructure risks can be reduced by combining cloud computing and agile operations. Because of its versatility, cloud computing can leverage several hardware and software platforms. Reducing expensive software and hardware requirements reduces application development costs. This function allocates resources dynamically based on demand.

Project failure often results from poor team communication. This can be fixed using cloud computing. There are many Software-as-a-Service (SaaS) alternatives that improve cooperation by sharing resources and data.

Early in development, team members' priorities and goals may differ, making communication difficult. Each iteration is much faster when group models are shared, reviewed, and updated using cloud storage.

Cloud computing can efficiently manage an application's lifecycle. These technologies are suitable for cloud use because they can be readily integrated or used to construct cloud-based applications.

Testing usually takes place at the development team's office with customers. Bottlenecks result from a lack of operational program instances compared to site traffic. Cloud computing allows anyone with a stable Internet connection to use the software worldwide. The present codebase can be improved gradually in the cloud to enable for further cloud computing testing. The tester or consumer needs only the testing environment URL to access the cloud-based testing program. Consumers can provide feedback and review the change.

Longer feedback pathways reduce require misrepresentation and misinterpretation. Software development time and effort depend on client satisfaction. Cloud-based software makes end-user support easier. This accessibility makes it easy for users to brainstorm, describe their needs, and decide if they fit cloud computing.

Customers can request local workers to build

and distribute prototypes on a cloud-based platform for faster feedback. Unapproved prototypes remove risks compared to on-premise implementation, which demands large infrastructure investments.

Comparing the development of application when using agile methods with and without cloud computing

Figure 3 shows the project's agile development process, covering all iterations. Figure 3 shows the agile development phases and their durations. Step lengths, as seen in Figure 1. The above benefits significantly affect 1. 3.

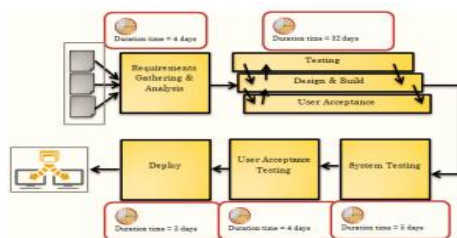


Figure 3: Cloud and agile development have been proved to reduce development times by days.

Diagram in Figure 1. Table 4 compares agile and cloud computing application development times.

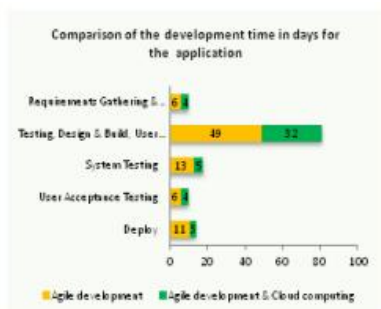


Figure 4: Combining cloud computing and agile methods drastically saves development time, improving efficiency. Longer stages are needed to truly benefit from cloud computing and agile development. The aforesaid combo has many benefits.

Without the client at the conference, cloud-based solutions like Skype were utilized to boost meeting participation and learn how software capabilities functioned. Google Forms and Docs simplified scientific paper sharing and poll participation.

The Eclipse IDE streamlines application development. Other Eclipse tools can be used to create code metrics or speed up repository integration.

Some instruments and versions are obsolete. The lack of cloud computing made software development challenging. Multiple tools' failure to communicate delayed the project's commencement until it was resolved. The job took two weeks longer than projected and cost extra.

Deploying a system via the cloud saves labor, time, and money compared to on-site installation.

Uploading the increment or whole program to Google Program Engine via the Google Eclipse Plugin validates it. Web browser users made program user feedback testing faster.

Because the second project iteration included cloud computing, an extra criterion was needed. Thus, more resources were needed. Cloud computing made change management and resource distribution easier, so the phase lasted the same amount of time.

5. CONCLUSION

This study examined the many benefits of combining cloud computing and agile software development as a new idea and improvement technique. Each selected company's resources and conditions are examined to determine why it adopted a cloud computing solution. A warehouse management application case study illustrates the suggested cloud computing technology and Dynamic Systems Development Method (DSDM) approach. Agile software development is tied to cloud-based agile. Openly stating the second strategy's benefits. Cloud computing, considered the future of IT, uses a web browser to download and install software. Agile software development methods allow frequent changes to software requirements and produce high-quality software by delivering and developing 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.
<http://dsdmofagilemethodology.wikidot.com/>
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/>
9. R. Shriver, Agile Cloud Development, The Virtualization Practice, LLC, 4. June 2012
<http://www.virtualizationpractice.com/agile-cloud-development-the-future-of-software-16226/>.