Jurnal Ilmiah Komputer dan Informatika
Edisi...Volume..., Bulan 20..ISSN :2089-9033

AUTOMATIC DEPLOYMENT SYSTEM USING CONTINUOUS INTEGRATION METHODOLOGY AT KAKATU

Jaga My Priera[1], Robi Tanzil Ganefi[2]

[1, 2] Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
Email : jagamyp@gmail.com[1], robi.tanzil.g@gmail.com[2]

ABSTRACT

PT. Kreatif Kinest Ideata or better known as Kakatu is a company in Information Technology (IT). Product and services PT. Kreatif Kinest Ideata is an Android parental control application named Kakatu. By using Kakatu, parents can control the use of gadgets by children. Based on the results of interviews with Kakatu, it's known that the Android application development process in Kakatu separated into several stages of the analysis, development, testing and deployment. In these stages, problems occurred in the development and testing process. Which at this stage the developers use third-party services such as Google Drive to distribute the apk file to the tester team, and Gmail as a medium of communication between the team of testers and developers. In the absence of an integrated system to perform both of these activities lead to difficult process of documenting the applications developed.

To solve these problems, author builds a Website Automatic Deployment using Continuous Integration Methodology(CI). CI is a software development method in which each developer integrates the application code they have made into a repository. The codes are then compiled and tested. Activities are triggered whenever developers commit to the repository. One of the popular CI software that is Jenkins. With various Jenkins plugins, allowing developer to build a CI system rich in features. One frequently used Jenkins plugin is a plugin to sending executable file via email automatically after the build process successfully.

Based on the analysis and testing that has been done by involving Kakatu, it can be deduced that this system has a decent used by the Kakatu to help speed up and simplify the process of software development Android.

Keywords : *Continuous Integration*, Jenkins, Kakatu, *Automatic Deployment*

1. PRELIMINARY

In this section will be explained the background of the problem, goals and objectives.

1.1. Background

PT. Kinest Kreatif Ideata or better known as Kakatu a startup company from Bandung. Kakatu was founded in 2014. Kakatu has two products and services those are Kakatu Launcher and Hyjabs. Kakatu is an Android parental control application. By using this app, parents can control the use of gadgets by children. While Hyjabs is a social media application to share the latest trends hijab. Through Hyjabs, users can share tips, trends, and tutorials about the use of the hijab.

Based on the results of interviews with Android developers at Kakatu author obtained some facts about Android application development process up to the release stage.At the first stage each developer writing code on their computers, then the code is compiled and tested, if there is no problem, all the changes the code will be stored (commit) to Bitbucket (push). After that the code will be merged (merge) and then compiled again, then the compilation result (apk file) will be sent to the testers group. If there is a problem the test team will report to the developer, if there are no problems then the application is ready to be released. The software used for the development of such applications is the Android Studio as IDE, Git as VCS, Bitbucket as management services repository, Gmail as a medium of communication between team members, and Google Drive or Dropbox as a medium for distributing the apk file to the team members who will perform testing applications. During the application development process found some problems, which is to share apk file to a team of testers, developers must upload the file to Google Drive or Dropbox beforehand. After the apk file is downloaded and tested, then the test team will provide feedback via email.

Other problems occur during the apk file documentation. In the previous paragraph has explained that the apk be tested shared via Google

Drive and Dropbox, while reports on the test results apk in Gmail. The differences in these locations causes problems when collecting information for documentation.

Based on the above description of the problem, then the solution offered is to build systems that can perform integrated code development process using Jenkins and documenting the results of the process by using Jenkins APIs. Author choose Jenkins as CI Tool because Jenkins has many plugins and supports all kinds of repositories, whether it Bitbucket, Gitlab, Github, or another kind of CVS such as Launchpad Bazaar. In addition, Jenkins can be installed on its own server, so users can build a CI system that fits their needs.

1.2. Purpose and Objectives

The purpose of this research is to build Automatic Deployment System Method Using Continuous Integration in Kakatu. While the objectives to be achieved are:

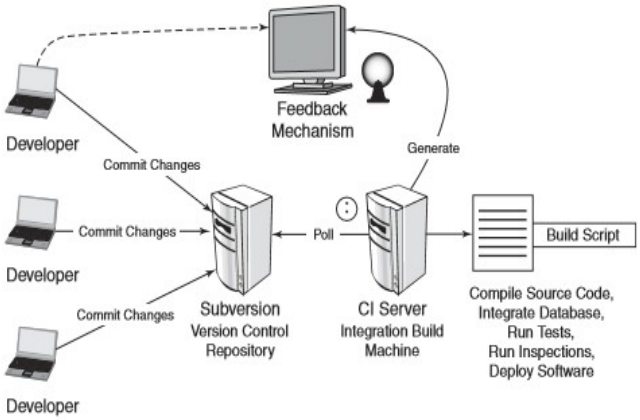
1. Facilitate the process of distributing the application to the tester group.
2. Facilitate communication between testers and developers
3. Streamlining apk and feedback documentation

1.3. *Continuous Integration (CI)*

CI is a method in software development, where each developer integrate and build their code at regular intervals to detect errors as early as possible.
In general, the stages in a CI system is as follows (triggered by the push command or commit):

1. Integrated
   Is the phase in which each developer integrate the codes or the results of their work.
2. Compiled
   Compiled code into executable
3. Tested
   Perform manual or automatic testing
4. Archived
   Archiving excutable, test results, and log during the process.
5. Deployed
   Installing or share the results of the build.

Here's a picture of a system architecture CI:



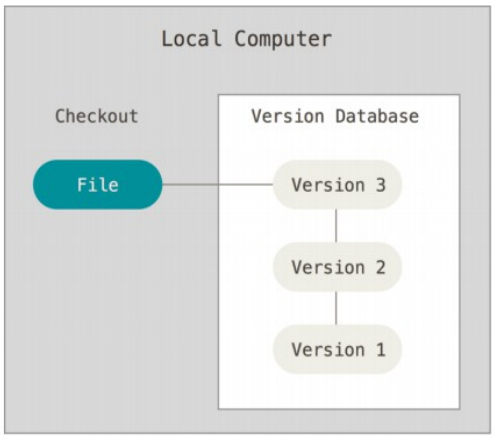
1.4. Jenkins

Jenkins is one of the CI tool developed by Kohsuke Kawaguchi with the first name Hudson in late 2004 when he was working at Sun. Hudson evolved over the years, this system was adopted by many teams in the Sun for their projects. In early 2008, Sun recognizes the quality and value of the Hudson, and ask Kohsuke to work on a full-time Hudson, and began to provide professional services and support Hudson. In 2010, Hudson has become the most preferred tool in Continuous Integration solutions with a market share of more than 70%.
In January 2011, the Hudson developer community decided to change the name of the project into Jenkins. They then make a copy of the original Hudson code to new GitHub project and continue with their work there. Most of the core and plugin developers along Kohsuke Kawaguchi and a key contributor jointly began developing Jenkins.

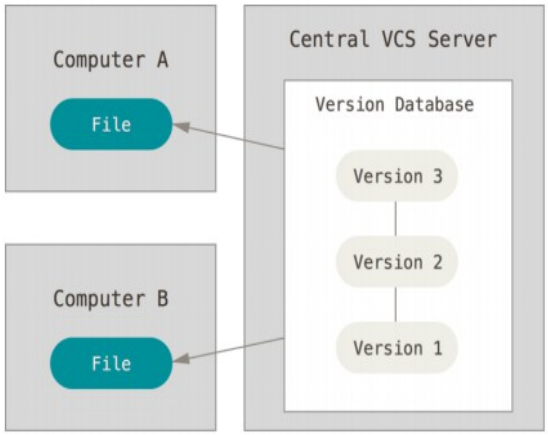
1.5. Version Control System (VCS)

Version Control System is a system that records changes to files from time to time. For example, a web designer will keep every version of the draft were made. By using VCS allows it to return to the previous design file, comparing the changes, and see who has made changes.

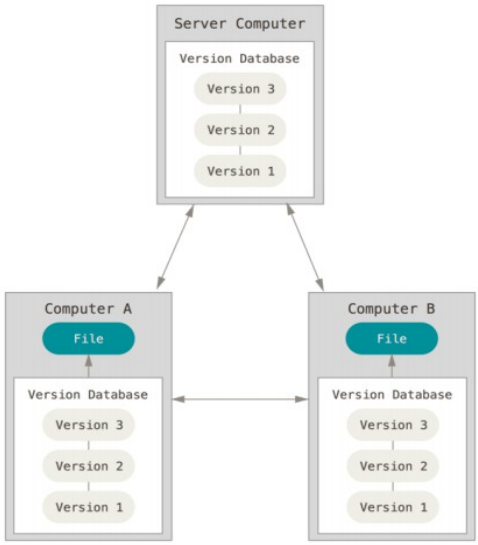
1. Local Version Control Systems (LCVS)



2. Centralized Version Control Systems (CVCS)



3. Distributed Version Control Systems (DVCS)



2. MAIN CONTENT

2.1. Design and System Analysis

In this section will explained about the analysis and design of system

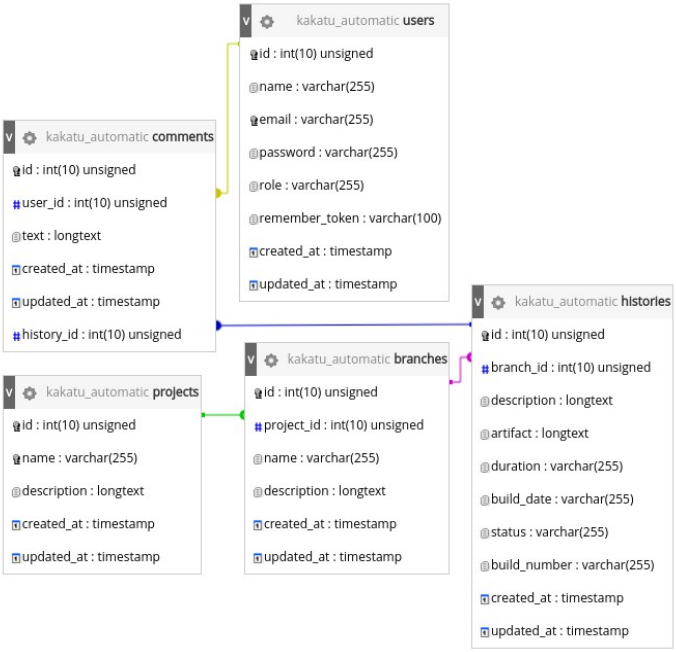
2.1.1. Problem Analysis

Below are the details of the problem that occurs on the current running system:

1. The process of compiling the code on each computer may causes problems.
2. Apk files shared via Google drive or Dropbox requires extra steps, such as uploading apk file manually.
3. Notification or notes about problems that occur in applications send via email (Gmail) requires extra steps, such as attaching a file version apk problematic.
4. Process documentation apk hard to do. Because the apk files and records on the apk file is in a different location.

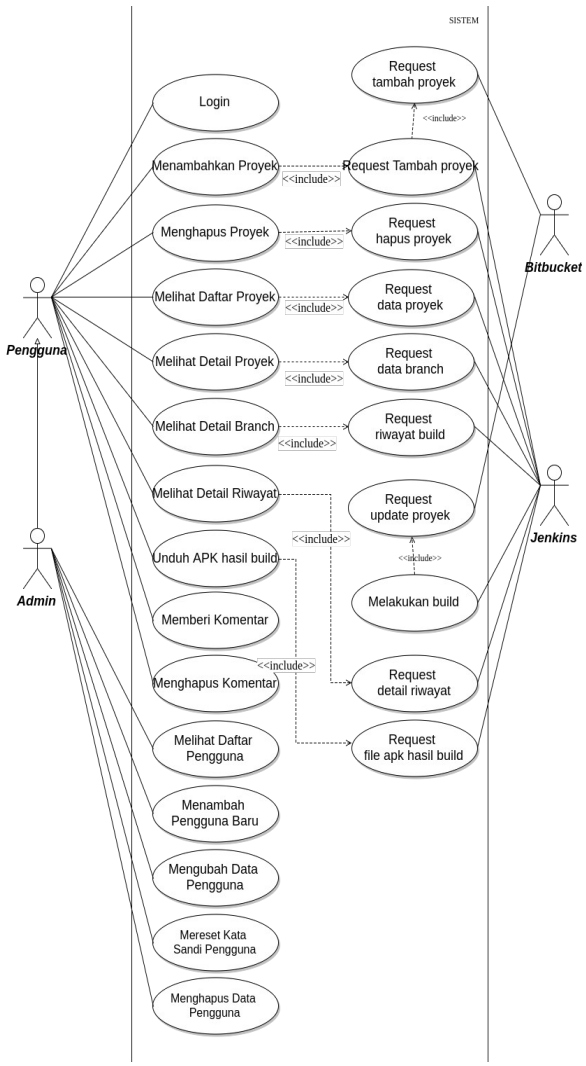
2.1.2. Data Analysis

Relation scheme of system:

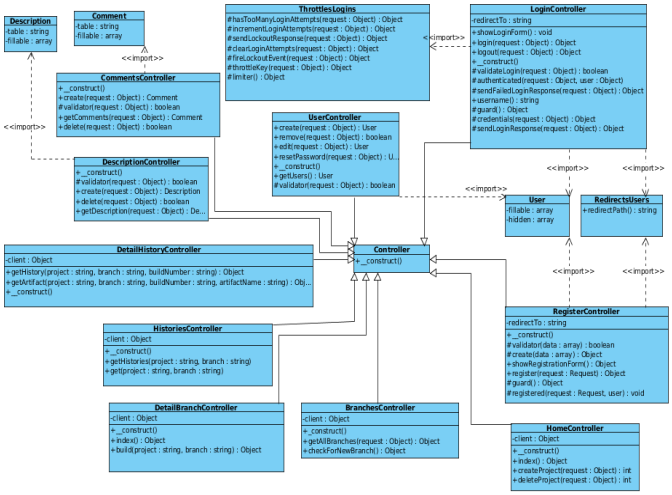


2.1.3. Functional Needs Analysis

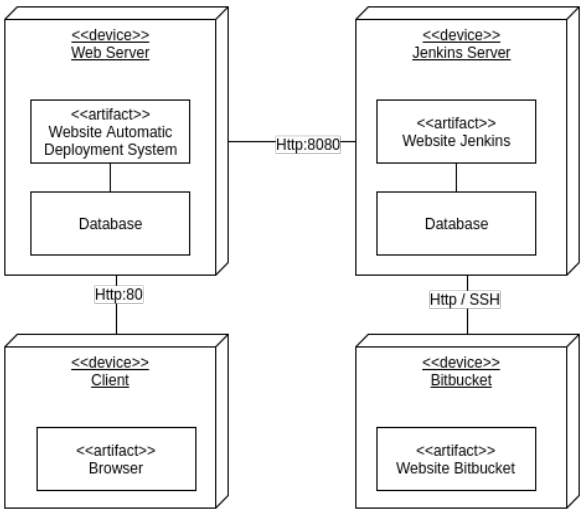
Use case diagram of system :



Berikut adalah diagram kelas sistem:

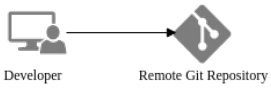
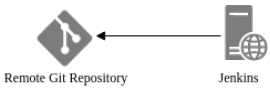


Berikut adalah diagram deployment sistem:



2.1.4. Build Process Analysis

Jenkins will check every latest commit on the remote repository on a regular basis. But the build process will only be executed if there is a keyword "[ci build]" in the commit message.

Git Commnad	Actor	Diagram
git commit -m "[ci *build*]"	*Developer*	
git fetch	Jenkins	

2.2. Implementation and System Testing

In this section will explained about the implement of system and the testing results

2.2.1. Implementation Environtment

Implementation environment explains about the specification of hardware, software and servers where the system will be installed and accessed. Here is an implementation environment:

1. Hardware Environment

Hardware specifications that are used are as follows:

Table 2.1: Hardware Environment

No	Item	Specification
1	Processor	Intel Celeron N2380 2.4Ghz
2	SSD	128 GB
3	Memory	8 GB
4	Monitor	LCD 11,6”
5	Internet Connection	Present
6	Mouse	Logitech M325
7	Keyboard	*On board*

2. Software Environment

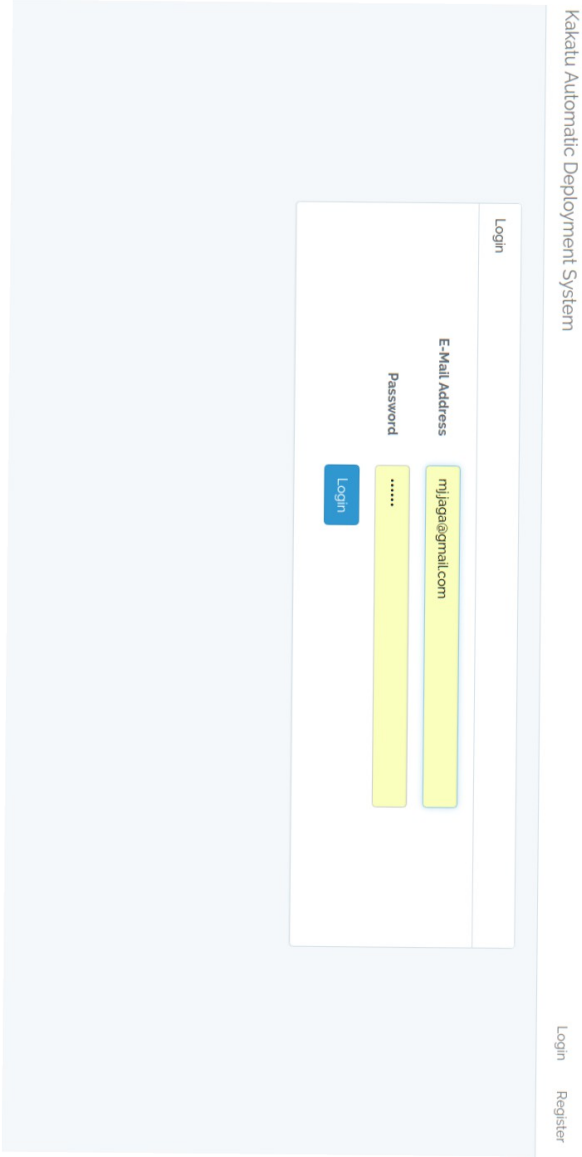
Software specifications that are used are as follows:

Table 2.2: Software Hardware Environment

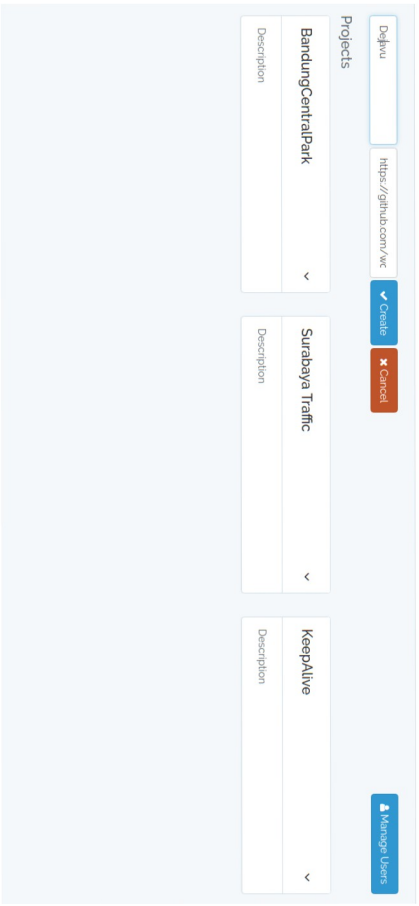
No	Item	Specification
1	Operating System	Elementary Loki 0.4
2	Browser	Google Chrome 56.0.2924.87
3	IDE	Sublime Text Build 3126
4	Programming Language	PHP 7.0 dan Javascript
5	Framework	Laravel 5.0 dan Angular
6	DBMS	Mysql v5.7.17
7	CI Tools	Jenkins v2.32.2
8	Web Server	Apache v2.4.18

2.2.2. User Interface Implementation

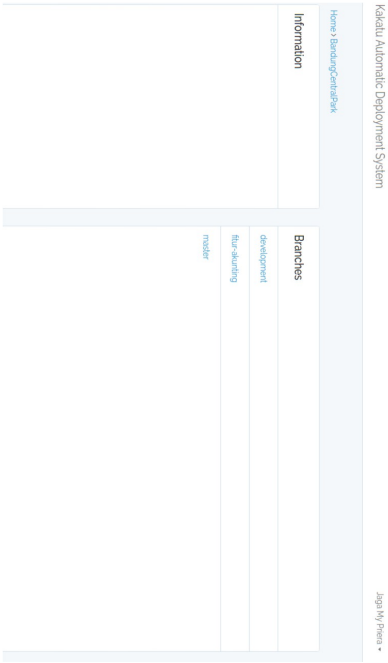
1. Login Page



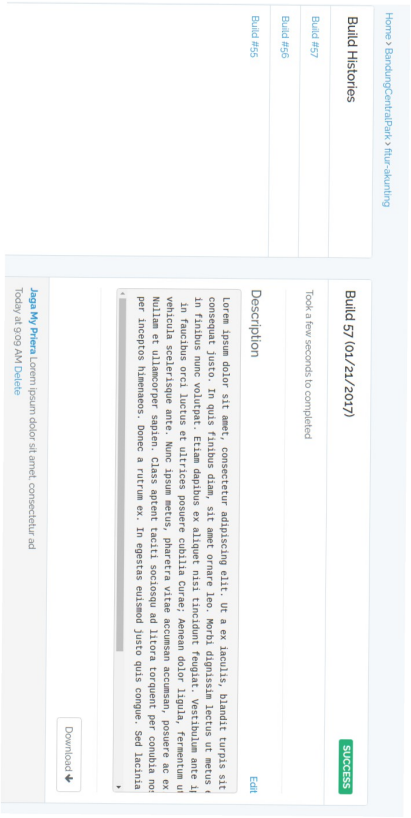
2. Project List Page



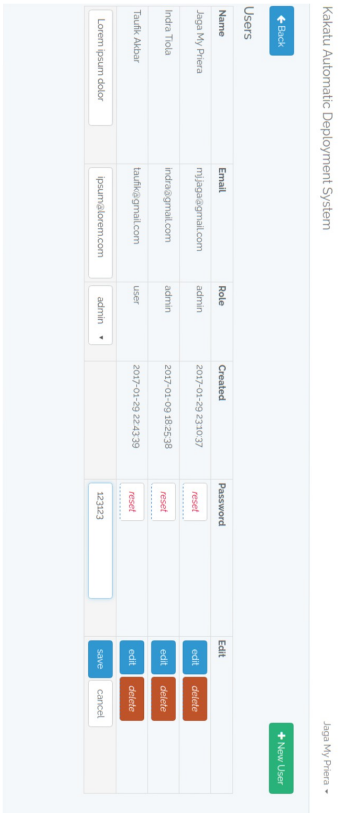
3. Branch List Page



4. Detail History of Build Page



5. User Management Page



2.2.3. Testing

After perform several blackbox tests, below are the comparison of this system and the old one.

Task	Old System	This System
Apk file distribution	Apk file uploaded to Google Drive or Dropbox, then share the download link via email	Apk file already available on the CMS when the developer commit code with build message
Feedback and Communication	Feedback and communication is done via email.	Feedback and communication is done in the CMS to the relevant version of the apk.
Documentation	Apk files collected from Google Drive and Dropbox. Feedback related to the test results are collected via email	Apk file and feedback results of test have been arranged according to each apk version.

3. CONCLUSION AND ADVICE

Conclusions of this software development are:

1. Automatic Website Deployment can speed up the distribution process to the testers in Kakatu.
2. Automatic Website Deployment can facilitate communication between developers and testers in Kakatu.
3. Automatic Website Deployment can simplify the process of documenting apk file and feedback in Kakatu.

The suggestions for the future development of this system are:

1. Develop this system into a service that can be used by other companies.
2. Adding features to automatically deploy to Google PlayStore.
3. Develop a mobile version of the application of this system

4. BIBLIOGRAPHY

[1] J. Ferguson Smart, *Jenkins - The Definitive Guide*, Creative Commons Edition ed. Wakaleo Consulting, 2011.

[2] S. Chacon and B. Straub, *Pro Git* in *Git - book*, Second Edition ed. Appress, 2014.

[3] R. Miles, Learning UML 2.0, K. Hamilton, Ed. O'Reilly, 2006.

[4] "Pengenalan JSON," dalam *JSON*. [Online]. Available: http://www.json.org/json-id.html. Accessed: 26 Agustus 2016.

[5] Gladwin. B, Ramya. R, dan Rathika. K, "Online Student Resource Management Using HTML5 and Angular -JS," *International Journal of Computer Science and Information Technology Research*, vol. 3, no. 1, hal. 53–57, Jan. 2015.

[6] V. Traynor, "What is the design sprint method and why do I think it's interesting? – Google developers experts," Medium, 2016. [Online]. Available: https://medium.com/google-*developer*-experts/what-is-the-design-sprint-method-and-why-do-i-think-it-s-interesting-3d51d5c37ac1. Accessed: Dec. 5, 2016.

[7] R. Susanto, and A. Dara Andriana, "Perbandingan Model Waterfall dan Prototyping untuk Pengembangan Sistem Informasi," *Majalah Ilmiah UNIKOM*, vol. 14, no. 1, pp. 41–46.

[8] "Google drive - Penyimpanan Awan & Pencadangan *file* untuk Foto, Dokumen & Banyak lagi,". [Online]. Available: https://www.google.com/intl/id/drive/. Accessed: Oct. 26, 2016.

[9] "Buat spreadsheet yang canggih" 2014. [Online]. Available:

https://www.google.com/intl/id/sheets/about/. Accessed: Oct. 26, 2016.
[10] Gmail, "Gmail,". [Online]. Available: https://www.google.com/intl/id/mail/help/features. html. Accessed: Oct. 26, 2016.