

Rancang Bangun Aplikasi Code Sharing Sebagai Alat Bantu Media Interaktif Perkuliahan Pada Mata Kuliah Pemrograman Web

Ar-Razy Muhammad¹, Heri Priyanto², Novi Safriadi³

Program Studi Teknik Informatika, Jurusan Teknik Elektro, Universitas Tanjungpura^{1,2,3}

e-mail: arrazymuhammad@live.com, heripriyanto.stmt@untan.ac.id, bangnops@gmail.com

Abstrak— Proses pengembangan aplikasi berbasis *web* dapat di definisikan sebagai proses pembuatan aplikasi dari yang hanya berupa satu halaman *web* hingga aplikasi kompleks dengan menghubungkan banyak halaman *web*. Proses pembelajaran pada mata kuliah pemrograman *web* kadang dibuat berkelompok agar mahasiswa dapat bekerja sama untuk menyelesaikan tugas yang lebih berat, seperti membuat aplikasi berbasis *web* yang kompleks, sehingga mahasiswa akan membuat banyak *file-file* program. Hal ini memungkinkan anggota kelompok untuk membuka dan mengubah data pada *file* yang sama dalam waktu bersamaan, namun dikarenakan ruang lingkup kerja yang berbeda maka akan ada ketimpangan data antar anggota tim. Sehingga membutuhkan penyesuaian lebih lanjut atas data tertentu agar setiap anggota memiliki data yang sama persis. Oleh karena ini *codesharing* dibuat sebagai sebuah aplikasi yang mampu menciptakan satu lingkungan kerja bagi anggota tim untuk mendapatkan data terbaru dari *file-file* yang dibutuhkannya secara *realtime*. Dikarenakan sifatnya yang *realtime* maka aplikasi ini dapat dimanfaatkan bagi dosen untuk dijadikan alat bantu media ajar pada proses perkuliahan. Aplikasi *codesharing* memfasilitasi pengguna untuk menciptakan sebuah *project* pribadi maupun bersama. Pengguna aplikasi dibedakan menjadi dua yaitu dosen dan mahasiswa. Setiap pengguna diberikan hak akses penuh atas *project* maupun *file-file* yang dibuatnya. Pengguna lain diberikan hak akses khusus yaitu dapat mengubah konten dari *file-file* apabila telah mendapatkan izin akses dari pemilik *file*. Dari hasil pengujian maka didapatkan bahwa aplikasi *codesharing* mendapatkan penilaian dari *user* dengan hak akses dosen sebesar 19 berdasarkan aspek penilaian Rekayasa Perangkat Lunak, 21 berdasarkan aspek penilaian Komunikasi Visual dan 53 berdasarkan aspek Fungsionalitas. Berdasarkan penilaian diatas maka aplikasi termasuk kategori baik. Penilaian *user* hak akses mahasiswa didapat nilai rata-rata sebesar 21.9 berdasarkan aspek penilaian Rekayasa Perangkat Lunak, 28.55 berdasarkan aspek penilaian Komunikasi Visual dan 47.55 berdasarkan aspek Fungsionalitas. Berdasarkan penilaian diatas maka aplikasi termasuk kategori baik. *user* menyatakan aplikasi sangat baik apabila diterapkan sebagai media ajar.

Kata Kunci— *collaborative code editor*, pemrograman berbasis *web*, *realtime application*

I. PENDAHULUAN

Pemrograman *Web* merupakan mata kuliah yang diberikan pada semester empat di Prodi Teknik Informatika Fakultas Teknik Universitas Tanjungpura. Pada mata kuliah ini mahasiswa belajar proses pengembangan aplikasi berbasis *web*. Pengembangan aplikasi berbasis *web* dapat di definisikan

sebagai proses pembuatan aplikasi dari yang hanya berupa satu halaman *web* hingga aplikasi kompleks yang menghubungkan banyak halaman *web*.

Proses pembelajaran pada mata kuliah pemrograman *web* kadang kala mewajibkan mahasiswanya untuk membuat kelompok. Tujuan pembentukan kelompok ini adalah agar mahasiswa dapat bekerja sama untuk menyelesaikan tugas yang lebih berat. Contohnya adalah membuat aplikasi berbasis *web* yang kompleks, sehingga mahasiswa akan membuat banyak *file-file* program. Hal ini memungkinkan dua orang atau lebih anggota kelompok membuka satu *file* program secara bersamaan. Permasalahannya adalah setiap anggota bekerja secara lokal, akibatnya akan ada perbedaan kode program pada saat aplikasi tersebut digabungkan. Sehingga pada saat aplikasi dijalankan, aplikasi tidak berjalan secara semestinya.

Permasalahan ini dapat diatasi apabila kelompok tersebut dapat bekerja di dalam sebuah lingkungan kerja yang memungkinkan tiap anggota kelompok bekerja secara bersamaan. Sehingga setiap anggota kelompok bisa mendapatkan pembaruan dari *file-file* yang diaksesnya secara *realtime*.

Aplikasi yang dapat menyediakan lingkungan kerja semacam ini diharapkan mampu dijadikan solusi dalam pengembangan *web* secara tim. Selain itu, aplikasi juga diharapkan mampu dimanfaatkan sebagai alat bantu perkuliahan bagi dosen dan mahasiswa sehingga dosen dan mahasiswa dapat berinteraksi secara aktif di dalam lingkungan kerja yang sama.

II. URAIAN PENELITIAN

A. Node.js

Mengutip dari laman resmi *node.js* [1] “*Node.js* adalah sebuah platform yang digunakan untuk memudahkan dalam membangun jaringan aplikasi yang bersekala dan cepat. *Node.js* menggunakan *event-driven, non-blocking I/O model* yang membuatnya ringan dan efisien, cocok untuk data-intensif dan aplikasi *real-time* yang berjalan di perangkat terdistribusi”

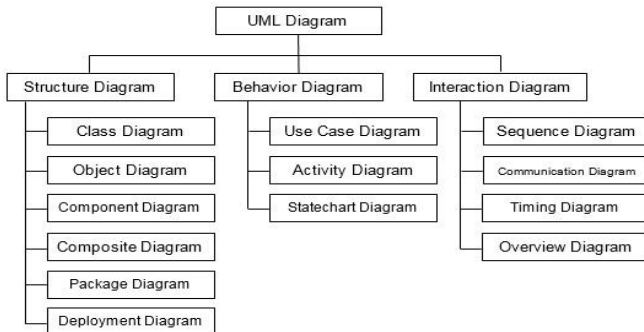
Node.js memungkinkan Javascript yang selama ini digunakan sebagai *client-side script* untuk bisa dijalankan di sisi server. *Node.js* memiliki pustaka *server HTTP* sendiri sehingga bisa menjalankan *webserver* sendiri.

B. CodeIgniter

Riyanto [2] menyatakan bahwa “CodeIgniter (selanjutnya disebut CI) adalah sebuah *framework* yang digunakan untuk membuat sebuah aplikasi berbasis *web* yang disusun dengan menggunakan Bahasa PHP. Didalam CI ini terdapat beberapa macam kelas yang berbentuk *library* dan *helper* yang berfungsi untuk membantu pemrogram dalam mengembangkan aplikasinya”.

C. Unified Modeling Language

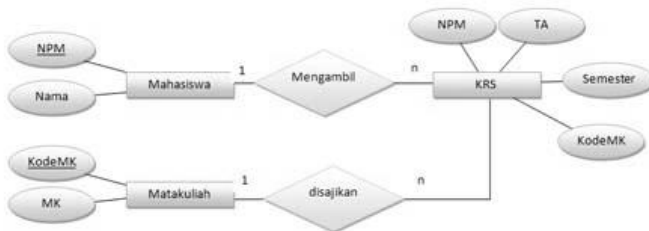
Unified Modelling Language (UML) adalah standard pemodelan dalam pembuatan aplikasi berorientasi obyek yang diajukan oleh *Object Management Group* (OMG) pada tahun 1996. Dalam UML terdapat 13 diagram yang dikelompokkan dalam 3 kategori [3]. Gambar 1 merupakan klasifikasi dari diagram UML.



Gambar 1. Klasifikasi Diagram UML

D. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan suatu model untuk menjelaskan hubungan antar data dan entitas dalam basis data berdasarkan object-object dasar data yang mempunyai hubungan antar relasi. ERD digunakan untuk memodelkan struktur data dan hubungan antar data [3]. Gambar 2 merupakan contoh *Entity Relationship Diagram*.



Gambar 2. Contoh *Entity Relationship Diagram*

E. Kuesioner

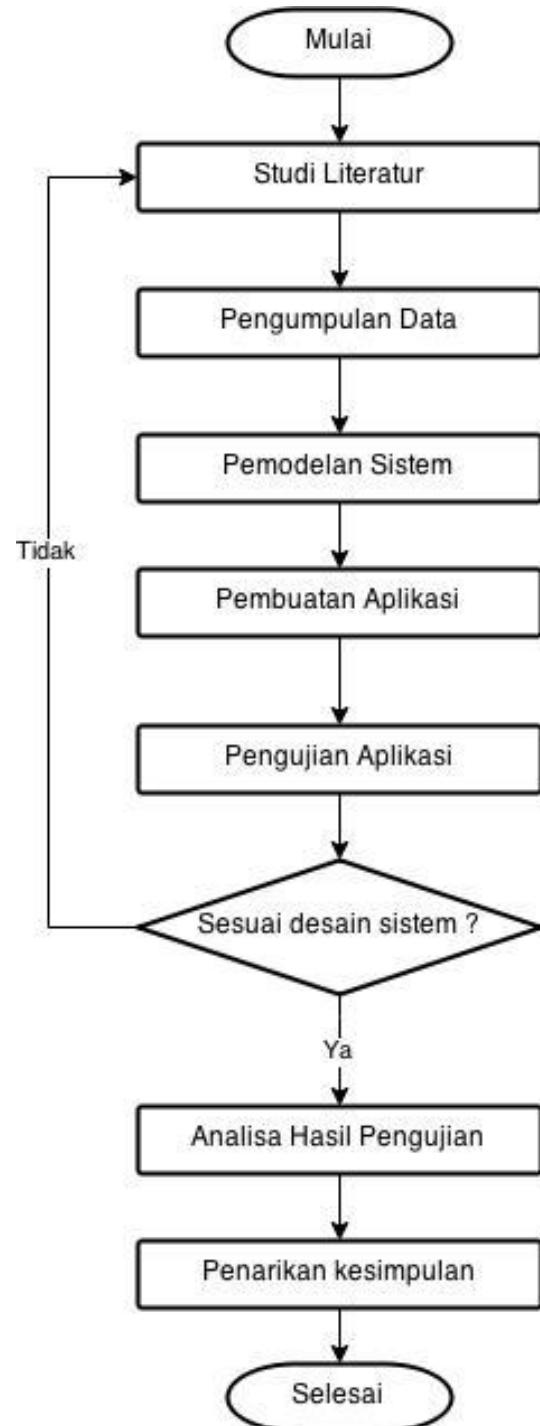
Dibandingkan dengan wawancara, kuesioner lebih efisien karena menuntut waktu lebih pendek dan biayanya pun lebih murah, sehingga memungkinkan pengumpulan data dari sampel yang lebih besar. Kuesioner adalah alat pengumpulan data yang lebih sering digunakan di dalam penelitian sosial [4].

F. Skala Likert

Skala Likert digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau sekelompok tentang kejadian atau gejala sosial. Dalam penelitian gejala sosial ini telah ditetapkan secara spesifik oleh peneliti, yang selanjutnya disebut sebagai variabel penelitian [5].

G. Diagram Alir Penelitian

Langkah-langkah yang dilakukan dalam penelitian dilakukan seperti pada Gambar 3.

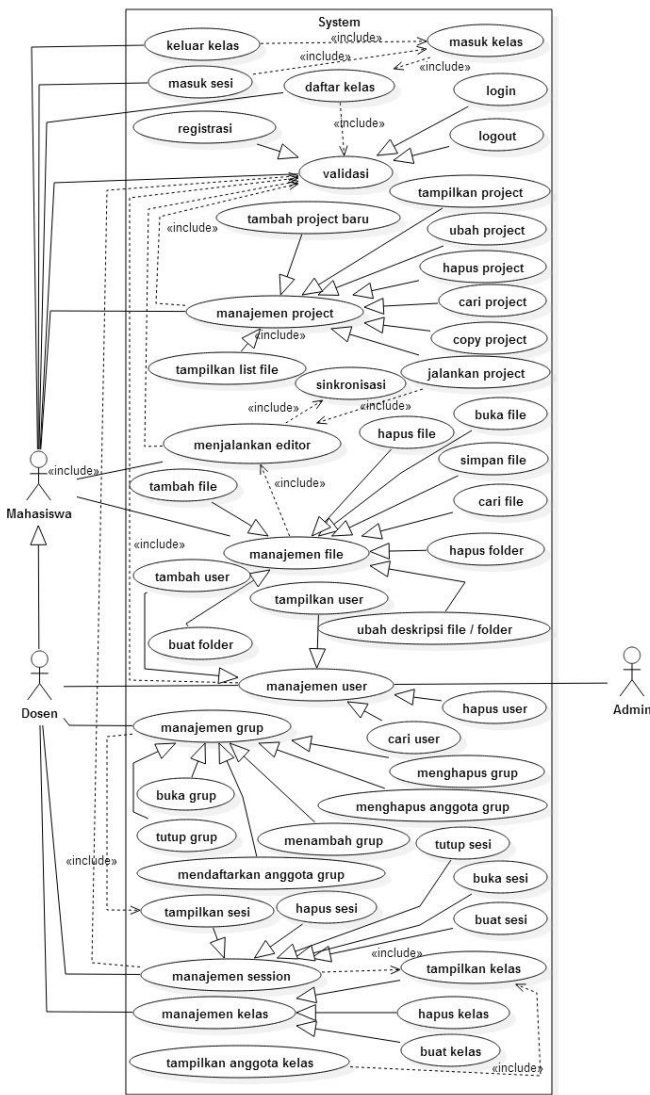


Gambar 3. Diagram Alir Penelitian

H. Perancangan Aplikasi

1. Use Case Diagram

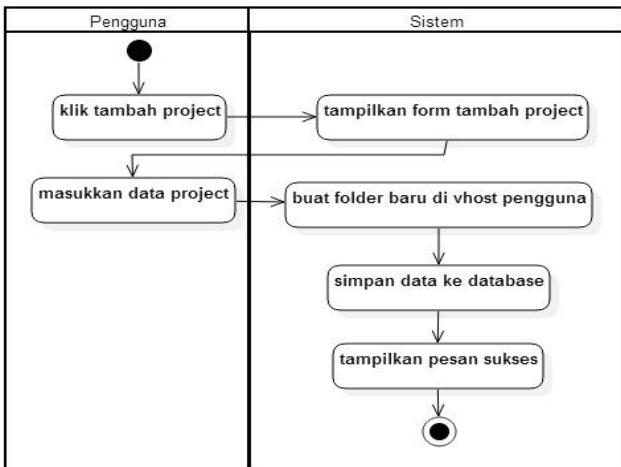
Use case diagram merupakan pemodelan untuk kelakuan system yang dibangun. *Use case diagram* menggambarkan interaksi antara satu atau lebih actor terhadap sistem. Gambar 4 merupakan *use case diagram* aplikasi yang dibangun.



Gambar 4. Use Case Diagram

2. Activity Diagram

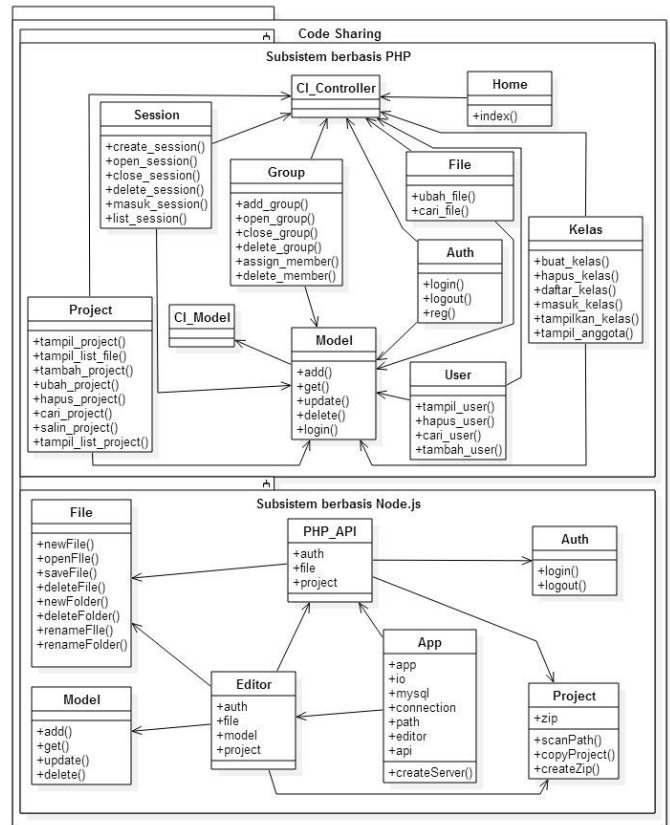
Diagram aktivitas menggambarkan aktivitas yang dilakukan sistem bukan aktivitas yang dilakukan oleh pengguna. Gambar 5 merupakan diagram aktivitas dari proses buat project.



Gambar 5. Activity Diagram buat project

3. Class Diagram

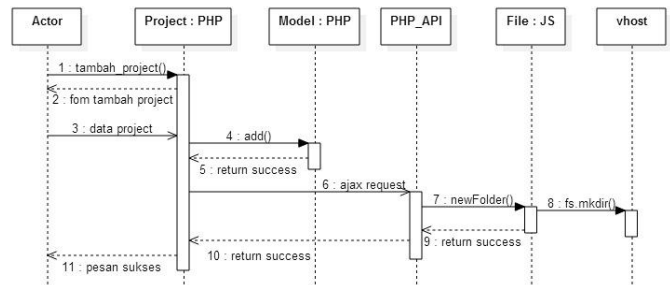
Diagram kelas menggambarkan struktur sistem berorientasi obyek dari segi pendefinisian kelas. Setiap kelas memiliki atribut dan metode atau fungsi yang digunakan dalam berjalannya sistem. Gambar 6 merupakan diagram kelas dari aplikasi.



Gambar 6. Class Diagram

4. Sequence Diagram

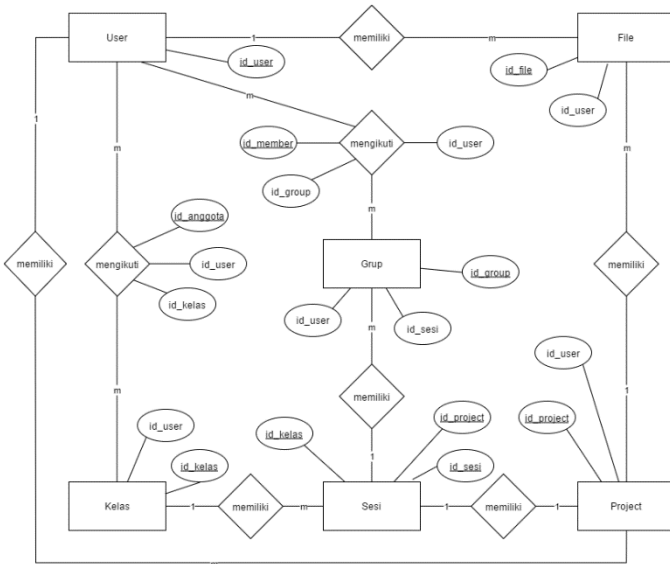
Sequence diagram menggambarkan kelakuan object pada usecase dengan mendeskripsikan waktu hidup obyek dan pesan yang dikirim dan diterima antar obyek. Gambar 7 merupakan sequence diagram dari proses buat project baru.



Gambar 7. Sequence diagram buat project

5. Entity Relationship Diagram

Entity Relationship Diagram adalah suatu model yang menggambarkan hubungan antar entitas yang digunakan dalam basisdata. Gambar 8 adalah model ERD yang digunakan untuk merancang basisdata aplikasi.



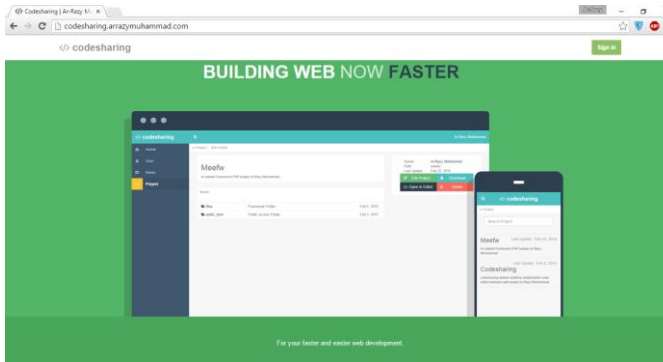
Gambar 8. Entity Relationship Diagram

III. HASIL DAN ANALISA

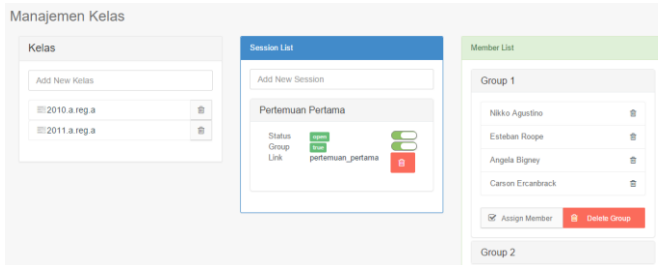
A. Hasil Implementasi

Aplikasi *code sharing* memerlukan *web server* untuk dapat dioperasikan. *Web Server* memerlukan ruang *haddisk* minimal 50 MB, RAM 4GB, aplikasi Apache 2, PHP5, MySQL dan Node.js. Selanjutnya diperlukan *web browser* yang mendukung HTML 5 untuk mengakses aplikasi.

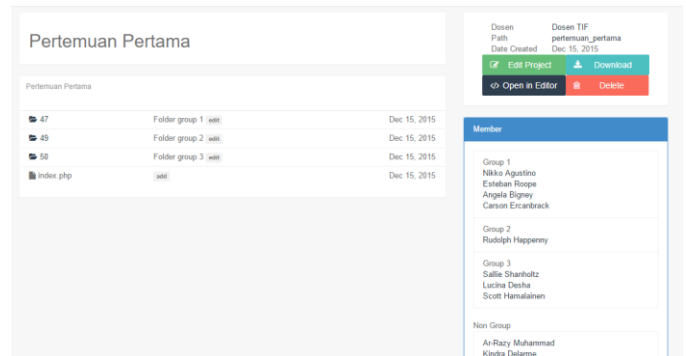
Tampilan antarmuka aplikasi dapat dilihat pada Gambar 9, Gambar 10, Gambar 11 dan Gambar 12.



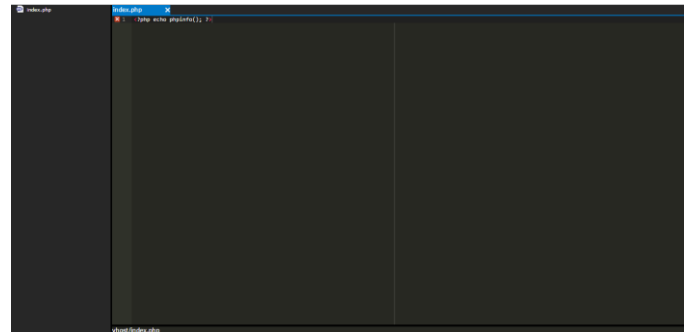
Gambar 9. Tampilan Antarmuka Halaman Landing Page



Gambar 10. Tampilan Antarmuka Halaman Manajemen Kelas



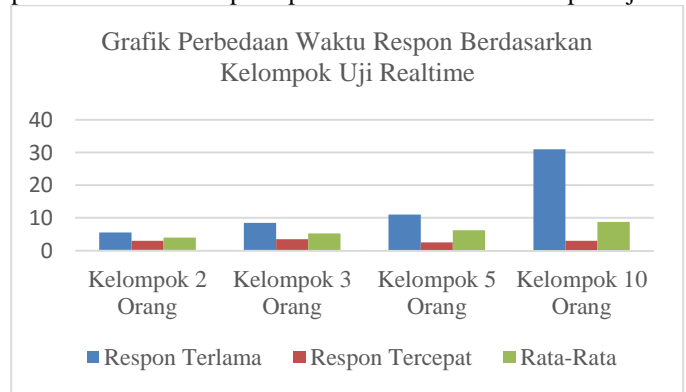
Gambar 11. Tampilan Antarmuka Halaman Project



Gambar 12. Tampilan Antarmuka Halaman Editor

B. Hasil Pengujian Realtime

Pengujian *realtime* merupakan pengujian yang dilakukan untuk mengetahui seberapa cepat pengguna lain menerima pembaruan data. Pengujian ini dilakukan dengan menyisipkan kode program untuk mencatat waktu data dikirim dan kemudian mencatat waktu data diterima. Perbedaan waktu ini dijadikan patokan seberapa cepat data diterima pengguna lain. Pengujian dilakukan dengan mencoba aplikasi di beberapa kelompok uji berbeda. Gambar 13 merupakan grafik perbandingan perbedaan waktu respon aplikasi berdasarkan kelompok uji.



Gambar 13. Grafik perbedaan waktu respon aplikasi

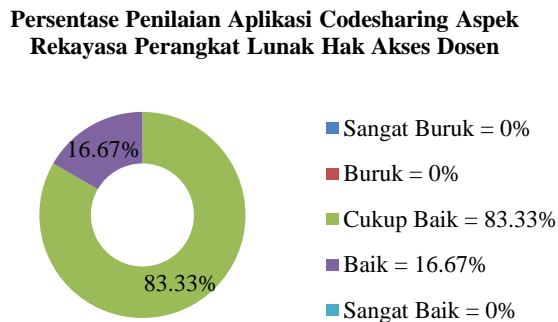
C. Hasil Pengujian Kuesioner

Pengujian Kuesioner dilakukan untuk mengetahui respon pengguna terhadap aplikasi *codesharing*. Pengujian dilakukan dengan mencoba aplikasi ke kelas Aplikasi Sistem Berbasis Web (ASBW) tahun ajaran 2015/2016 pada tanggal 7 Januari 2016 bertempat di Ruang A Gedung Teknik Informatika Universitas Tanjungpura.

Pengujian dibagi menjadi dua kelompok pengguna yaitu pengguna dengan hak akses Dosen dan Mahasiswa. Pengujian di tiap kelompok uji terbagi lagi menjadi tiga aspek pengujian, yaitu aspek pengujian Rekayasa Perangkat Lunak, aspek pengujian Komunikasi Visual dan aspek pengujian Fungsionalitas.

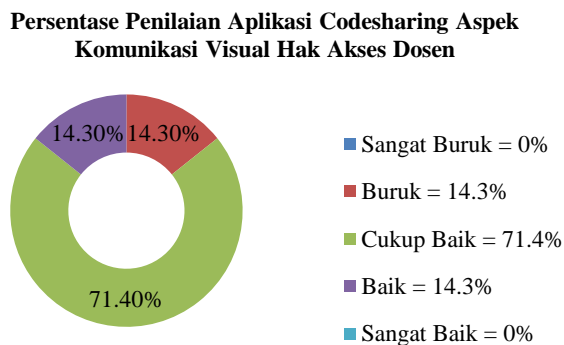
1. Pengujian Hak Akses Dosen

Gambar 14 merupakan hasil pengujian Aspek Rekayasa Perangkat Lunak dari pengguna dengan hak akses dosen.



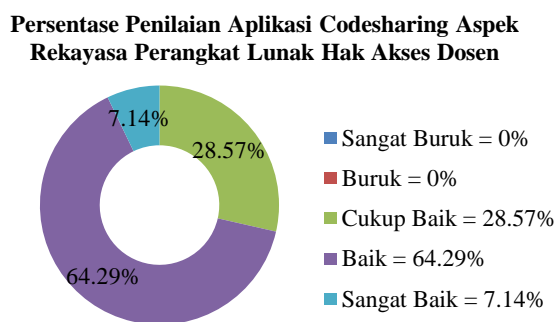
Gambar 14. Persentase penilaian Aspek RPL dosen

Hasil pengujian Aspek Komunikasi Visual dari pengguna dengan hak akses dosen dapat dilihat pada gambar 15.



Gambar 15. Persentase Penilaian Aspek Komunikasi Visual Dosen

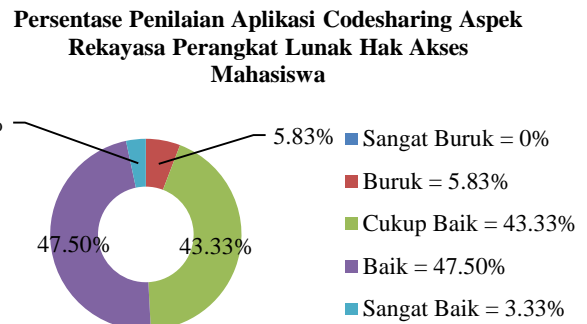
Gambar 16 merupakan hasil pengujian Aspek Fungsionalitas dari pengguna dengan hak akses dosen.



Gambar 16. Persentase Penilaian Aspek Fungsionalitas Dosen

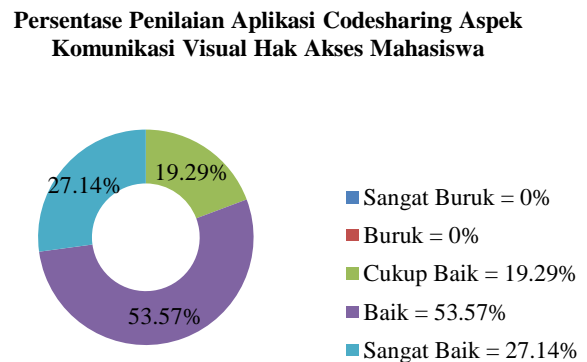
2. Pengujian Hak Akses Mahasiswa

Gambar 17 merupakan hasil pengujian Aspek Rekayasa Perangkat Lunak dari pengguna dengan hak akses mahasiswa.



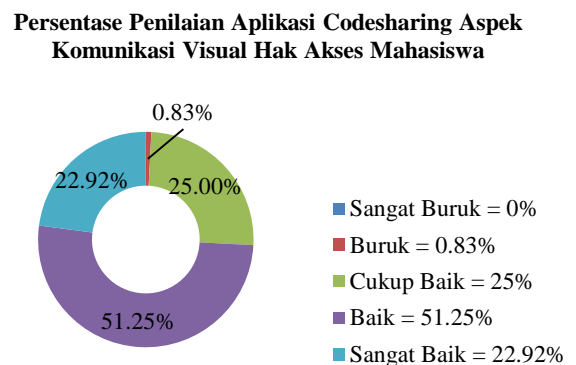
Gambar 17. Persentase penilaian Aspek RPL mahasiswa

Hasil pengujian Aspek Komunikasi Visual dari pengguna dengan hak akses mahasiswa dapat dilihat pada gambar 18.



Gambar 18. Persentase Penilaian Aspek Komunikasi Visual Mahasiswa

Gambar 19 merupakan hasil pengujian Aspek Fungsionalitas dari pengguna dengan hak akses mahasiswa.



Gambar 19. Persentase Penilaian Aspek Fungsionalitas Mahasiswa

IV. KESIMPULAN

A. Kesimpulan

Berdasarkan hasil analisa dan pengujian aplikasi *codesharing* maka dapat disimpulkan bahwa:

1. Aplikasi *codesharing* mampu menciptakan sebuah lingkungan kerja yang terintegrasi untuk pengembangan aplikasi berbasis *web*.
2. Aplikasi *codesharing* mampu memberikan waktu respon yang relatif cukup cepat jika dibandingkan dengan kedipan mata manusia.

B. Saran

Saran dari penulis untuk pengembangan aplikasi lebih lanjut adalah sebagai berikut:

1. Aplikasi dapat ditambahkan fitur *database* untuk setiap *user* sehingga *project* yang dibuat dapat langsung terhubung dengan basisdatanya masing-masing.
2. Aplikasi sangat memungkinkan untuk penggunaan yang lebih luas sehingga untuk pengembangan selanjutnya sangat disarankan untuk bisa memiliki fitur *version control*.

DAFTAR PUSTAKA

- [1] Node.js Fondation. 2015. *About Node.js* (<http://nodejs.org/en/about>, diakses pada tanggal 20 Maret 2015).
- [2] Riyanto. 2011. "Membuat sendiri aplikasi e-commerce dengan PHP & MySQL menggunakan CodeIgniter & JQuery". Yogyakarta: Penerbit Andi.
- [3] A.S, Rosa; Shalahuddin, M. 2013. "Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek". Bandung: Informatika.
- [4] Riduwan. 2008. "Metode dan Teknik Menyusun Tesis". Bandung: Alfabeta.
- [5] Sukadji, Soetarlinah. 2000. "Menyusun dan Mengevaluasi Laporan Penelitian". Jakarta: Penerbit Universitas Indonesia.