

PROPOSAL SKRIPSI

APLIKASI KALENDER UNTUK MENGELOLA JADWAL KEGIATAN DI UMS MENGGUNAKAN *GOOGLE APPLICATION API*



Diajukan Sebagai Salah Satu Syarat Menyelesaikan Jenjang Strata I
Pada Jurusan Teknik Informatika Fakultas Komunikasi dan Informatika
Universitas Muhammadiyah Surakarta

Disusun oleh:

SUYADI

L200100015

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA
OKTOBER 2013**

USULAN PENELITIAN SKRIPSI

APLIKASI KALENDER UNTUK MENGELOLA JADWAL KEGIATAN DI UMS MENGGUNAKAN *GOOGLE APPLICATION API*

Yang diajukan oleh:

Suyadi

L200100015

Telah disetujui oleh:

Pembimbing,

(Gunawan Ariyanto, ST, M.CompSc, Ph.D.)

NIK: 968

Pada tanggal:

A. Judul Usulan Penelitian

Aplikasi Kalender untuk Mengelola Jadwal Kegiatan di UMS Menggunakan *Google Application API*.

B. Latar Belakang Masalah

Pada saat ini, aplikasi kalender berbasis web sudah dipakai secara luas (Stenchikova et al. 2007). Banyak perguruan tinggi yang menggunakan aplikasi kalender berbasis web untuk mengelola berbagai kegiatan, misalnya University of Minnesota menggunakan kalender untuk menjadwalkan kegiatan akademik, membuat kalender pribadi, berbagi kalender dengan orang lain dan mengundang teman ke suatu acara (OIT 2013).

UMS sudah memiliki aplikasi penjadwalan online yaitu aplikasi Jadwal Terpadu. Aplikasi ini dirancang untuk membantu menyusun jadwal kuliah secara mudah dengan mengoptimalkan penggunaan sumber daya dosen dan ruang sambil menghindari adanya tabrakan jadwal bagi peserta kuliah. Aplikasi ini dapat diakses melalui intranet atau jaringan di dalam kampus (UMS 2012). Fitur dari aplikasi Jadwal Terpadu UMS masih sederhana, dosen dapat melihat jadwal mengajarnya, mahasiswa dapat melihat jadwal kuliahnya, dan semua orang dapat melihat jadwal kuliah berdasarkan jurusan, dosen, ruang dan mata kuliah tertentu, tetapi proses input jadwal masih dikerjakan secara manual.

Google Calendar adalah layanan kalender yang disediakan Google secara gratis bagi pelanggannya. Pemilik akun Google dapat membuat kalender, membuat acara dan mengundang orang lain ke dalam acara tersebut. *Google Calendar* dapat diakses melalui telepon genggam dan dapat memberi peringatan melalui SMS atau email. Google menyediakan *Application API*

yang memungkinkan pengembang untuk membuat aplikasi yang dapat mengakses aplikasi-aplikasi Google seperti Gmail, Calendar dan aplikasi lain menggunakan *Google Data APIs*, *Gadgets*, dan *Google Apps Script* (Google Developers 2013).

Berdasarkan uraian di atas, penulis mengusulkan pembuatan aplikasi kalender berbasis web menggunakan *Google Application API* untuk mempublikasikan jadwal kuliah sebagai pelengkap aplikasi Jadwal Terpadu yang telah ada dan menambah fitur untuk mengelola agenda kegiatan lainnya, seperti: seminar, rapat, dan pameran.

C. Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah, “Bagaimana membuat aplikasi kalender berbasis web untuk mengelola kegiatan dengan memanfaatkan *Google Application API* ?”

D. Batasan Masalah

Batasan masalah dalam penelitian yang akan dilakukan penulis agar lebih fokus pada masalah yang diteliti adalah:

1. Aplikasi dibuat dengan bahasa pemrograman Python, server database MySQL dan *Google APIs Client Library for Python*.
2. Aplikasi dapat diakses menggunakan *web browser* melalui internet.
3. Aplikasi digunakan untuk membuat kalender di *Google Calendar* pada domain *ums.ac.id*, membuat *event* di dalamnya, dan mengundang orang lain bergabung dalam *event* tersebut.
4. Aplikasi dilengkapi fasilitas import data dari aplikasi Jadwal Terpadu UMS.

5. Aplikasi dapat digunakan oleh semua orang yang memiliki akun *Google Apps* pada domain *ums.ac.id* dengan level *owner* dan *guest*. *Owner* dapat membuat, mengubah atau menghapus kalender dan *event*, *guest* dapat melihat dan mengubah *event* sesuai hak yang diberikan oleh *owner*.
6. Aplikasi dilengkapi fasilitas pencarian *event* dan menyalinnya ke dalam kalender pribadi.
7. Aplikasi dilengkapi fasilitas untuk membuat daftar tugas secara otomatis saat membuat *event*.
8. Fitur lain yang ditambahkan pada aplikasi ini adalah membuat surat undangan untuk *event* rapat menggunakan Google Docs yang dikirim ke alamat email pengguna yang diundang rapat.

E. Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Mengetahui fasilitas-fasilitas yang disediakan *Google APIs Client Library for Python* sebagai sarana komunikasi bahasa pemrograman Python dengan data-data *Google Apps*.
2. Membuat aplikasi yang dapat digunakan untuk mengelola kalender dan *event* dari *Google Calendar*.

F. Manfaat Penelitian

1. Bagi Pengguna

Manfaat aplikasi yang dihasilkan dari penelitian ini yang dapat diambil oleh pengguna adalah sebagai berikut :

- a. Memudahkan program studi dalam menginformasikan jadwal perkuliahan.

- b. Memudahkan pengguna untuk mengelola tugas dan jadwal kegiatan secara mandiri maupun bersama-sama.
- c. Memudahkan pengguna untuk menelusur berbagai kegiatan yang diselenggarakan di UMS.

2. Bagi Peneliti

Manfaat yang dapat diambil oleh peneliti adalah dapat mengembangkan ilmu yang diperoleh dari perkuliahan, meningkatkan kemampuan untuk membuat program khususnya dalam bahasa pemrograman Python, dan menambah wawasan dan ketrampilan di bidang teknologi informasi.

G. Tinjauan Pustaka

1. Telaah Penelitian

Maria Claudia Buzzi et al. (2010) meneliti interaksi *screen reader* dengan *Google Docs* untuk mengevaluasi peluang bagi orang buta untuk menulis dokumen kolaboratif dalam “Accessing Google Docs via Screen Reader”. Analisis menyebutkan adanya isu yang memiliki dampak negatif pada halaman muka *Google Docs* berkaitan dengan pemakaian antar muka bagi orang buta. Dia merumuskan saran dasar untuk meningkatkan interaksi dan membuat pengguna lebih puas, termasuk orang yang memiliki kekurangan.

Stenchikova et al. (2007) menjelaskan dalam “RavenCalendar: a Multimodal Dialog System for Managing a Personal Calendar” tentang pembangunan perangkat lunak *RavenCalendar*. *RavenCalendar* adalah sistem dialog *multimodal* untuk aplikasi *Google Calendar* yang memungkinkan pengguna membuat, memodifikasi dan menghapus *event* kalender, membuat query dan mendengarkan deskripsi *event*. Aplikasi ini

menggunakan *Google Maps* dan *Google Calendar* sebagai *Third-party backend server*.

Refanidis and Alexiadis dalam tulisannya “*SelfPlanner: An Intelligent Web-based Calendar Application*” menjelaskan pembangunan perangkat lunak yang *SelfPlanner* yaitu sebuah perangkat lunak yang dijalankan pada web server *dedicated*. Semua data pengguna tersimpan dalam server, kemudian pengguna dapat mengakses *SelfPlanner* melalui jaringan. Pengguna dapat membuat/mengubah data tugas menggunakan form yang *user-friendly*. Server memecahkan masalah penjadwalan dan menyisipkan entri ke dalam *Google Calendar* dari pengguna yang sesuai menggunakan *Google Api*. Pengguna mengakses langsung kalendernya di *Google Calendar*. Aplikasi dibangun dengan bahasa pemrograman Java.

2. Landasan Teori

Ada beberapa teori yang perlu dikaji dalam penelitian ini, yaitu pengertian Python, Django, MySQL, Web Server, Google Apps dan Google Application API.

a. Python

Python adalah bahasa pemrograman komputer *open-source* yang dioptimalkan untuk kualitas, produktivitas, portabilitas, dan integrasi. Python digunakan oleh ratusan ribu pengembang di seluruh dunia di berbagai bidang. Python mendukung pemrograman berorientasi objek. Python membuat proses pengembangan perangkat lunak menjadi lebih cepat dan menyenangkan (Mark Lutz 2001).

b. Django

Django adalah *framework* untuk membangun aplikasi web berbasis Python. Django menganut pola *Model-View-Controller* (MVC), model umumnya bertanggung jawab untuk mengelola data dan inti logik bisnis, *view* yang menampilkan data ke pengguna, *controller* menerima data dari pengguna dan melakukan logika secara spesifik terhadap aplikasi (Marty Alchin 2013).

c. MySQL

MySQL adalah *SQL database management system* yang cukup populer. MySQL merupakan database relasional yang menyimpan data dalam tabel-tabel terpisah dan dapat saling berhubungan. *SQL* bagian dari kata “MySQL” singkatan dari *Structured Query Language*. *SQL* adalah bahasa standar yang digunakan untuk mengakses database. Database Server *MySQL* sangat cepat, handal, terukur dan mudah digunakan. *MySQL* bekerja pada client/server atau pada sistem *embedded*. *MySQL* bersifat *open-source* (*MySQL 5.5 Reference Manual* 2013).

d. Web Server

Web server adalah sebuah server yang bertanggung jawab untuk menerima permintaan HTTP dari web client dan melayani respon HTTP kepada mereka, biasanya berupa halaman web statis (teks, gambar, dll.) dan konten dinamis (script). Salah satu web server yang sangat populer adalah Apache. Apache digunakan oleh lebih dari 50 % dari semua situs web (Igor Ljubuncic 2011).

e. Google Apps Platform

Médéric Morel et al. (2011) dalam bukunya *Google Apps: Mastering Integration and Customization* menjelaskan bahwa *Google Apps Platform* adalah sejumlah besar aplikasi dan layanan yang terintegrasi satu sama lain yang dikelompokkan dalam tiga kategori. Pertama, alat komunikasi yang terdiri dari Gmail, sebuah email sistem yang sekarang banyak dipakai secara luas, *Google Calendar* yaitu agenda online dengan fasilitas berbaginya, dan *Google Talk* yang digunakan untuk mengobrol secara online. Kedua, perangkat kolaborasi yang terdiri dari *Google Sites*, *Google Docs* dan *Address Book*. Terakhir, perangkat keamanan yang dikenal dengan *Postini Services*. Semua aplikasi dan service *Google Apps* dikaitkan dengan sebuah domain (misal: ums.ac.id). Layanan *Google Apps* tersedia dalam tiga edisi, yaitu:

1. *Standard Edition* untuk pengguna personal. Edisi ini gratis.
2. *Business Premier Edition* untuk perusahaan dengan harga langganan \$ 50 per user per tahun.
3. *Google Apps for Education* utamanya untuk universitas.

f. Google Application API

Google Application API memungkinkan pengembang aplikasi untuk menulis aplikasi yang mengakses aplikasi *Google* seperti *Gmail*, *Calendar* dan lainnya menggunakan *Google Data API*, *Gadget*, dan *Google Apps Script* (Google Developers 2013).

H. Metode Penelitian

1. Waktu dan Tempat

Waktu yang diperlukan untuk menyelesaikan penelitian ini adalah empat bulan, mulai bulan Oktober 2013 sampai dengan Januari 2014. Pengujian aplikasi dilaksanakan di Kantor IT Kampus II Universitas Muhammadiyah Surakarta.

2. Peralatan Utama dan Peralatan Pendukung

Peralatan utama yang diperlukan dalam penelitian ini adalah perangkat keras dan perangkat lunak. Perangkat keras yang digunakan untuk pengembangan aplikasi adalah laptop dengan spesifikasi sebagai berikut:

- a. Prosesor AMD E450 1.6GHz L2 Cache 1MB
- b. Hardisk 320 GB
- c. RAM 4 GB
- d. Sistem Operasi Debian 7 “Wheezy” 64 bit

Perangkat lunak:

- a. Python 2.7
- b. Database MySQL Server
- c. Paket-paket Python untuk web server, MySQL connector, dan Google Application API.
- d. Ninja IDE.

Adapun peralatan pendukung yang dibutuhkan sebagai berikut:

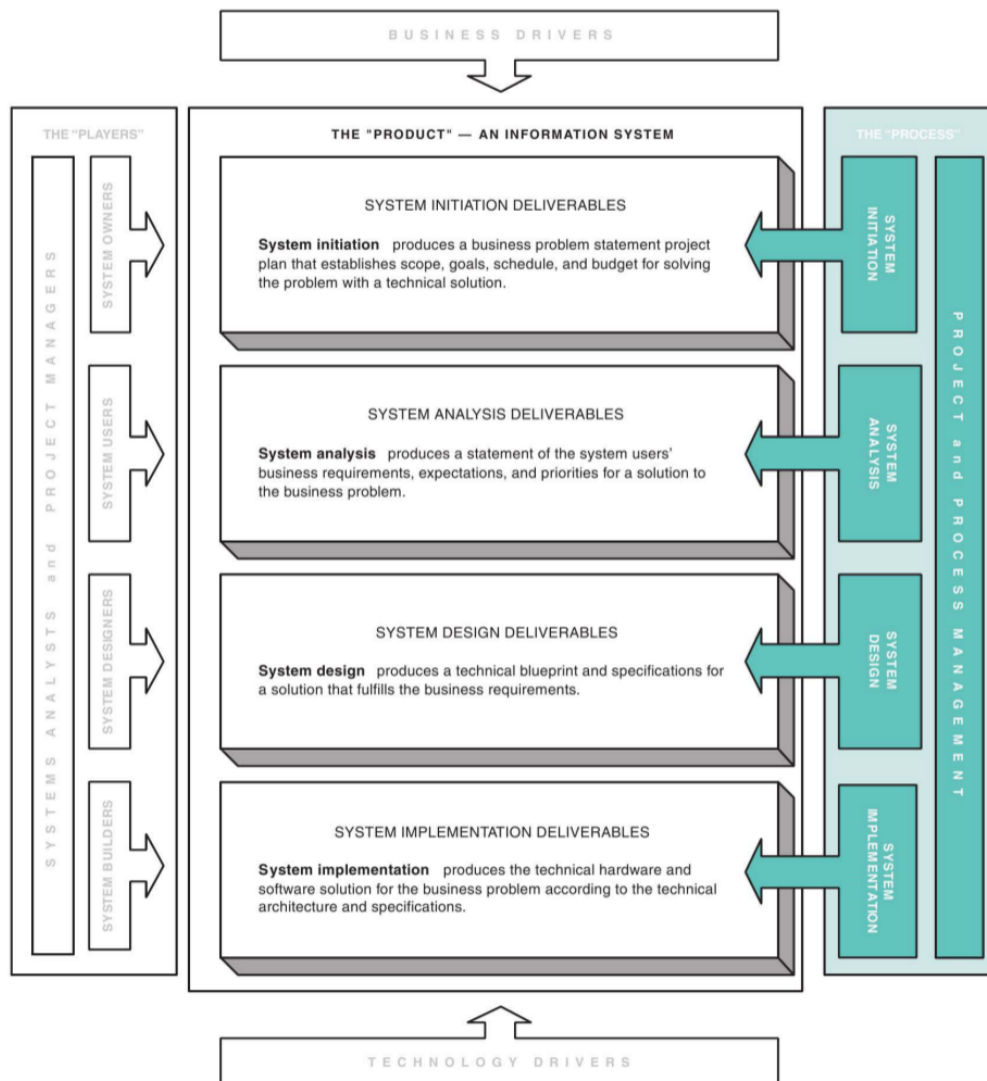
- a. Koneksi internet.
- b. Google Chrome web browser.
- c. Akun Google Apps domain ums.ac.id.

3. Diagram Alir Penelitian

Aktifitas dalam penelitian ini meliputi kegiatan rancang bangun aplikasi dan pembuatan laporan.

Pada proses rancang bangun, penulis menerapkan langkah-langkah manajemen proyek yang ditulis Jeffrey L. Whitten and Lonnie D. Bentley (2007) dalam bukunya “*Systems analysis and design methods*” untuk menghasilkan sistem informasi, yang meliputi: *system initiation*, *system analysis*, *system design* dan *system implementation*. Pada proyek yang sesungguhnya, ditambah aktifitas *system support and continuous improvement* dan masing-masing langkah memiliki penanggung jawab sendiri-sendiri.

Proses pengembangan dan pemecahan masalah digambarkan seperti bagan berikut:



I. Sistematika Penulisan Laporan

BAB I PENDAHULUAN

Mendeskripsikan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Berisi tentang teori-teori yang digunakan dalam penelitian, perancangan dan pembuatan sistem.

BAB III METODE PENELITIAN

Berisi metode-metode yang digunakan dalam perancangan aplikasi kalender yang dibuat.

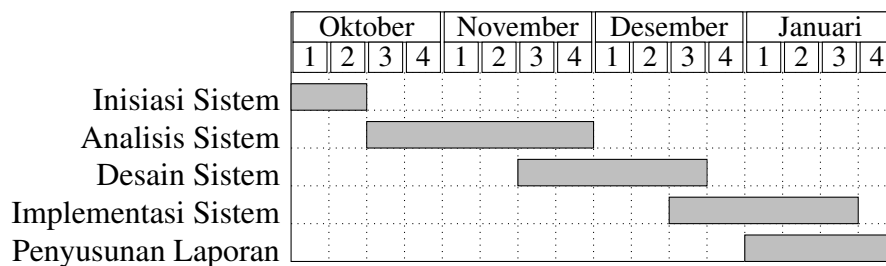
BAB IV HASIL DAN PEMBAHASAN

Memaparkan dari hasil-hasil tahapan penelitian, mulai dari analisis, desain, hasil testing dan implementasinya.

BAB V PENUTUP

Berisi kesimpulan dan saran dari seluruh penelitian yang telah dilakukan.

J. Rencana Kegiatan



K. Daftar Pustaka

Google Developers (2013). *Google Apps Application APIs (online)*. url: <https://developers.google.com/google-apps/app-apis>. Diakses: 2013-10-06.

Igor Ljubuncic (2011). *Apache Web Server Complete Guide*. Dedoimedo.

Jeffrey L. Whitten and Lonnie D. Bentley (2007). *Systems analysis and design methods*. 7th ed. McGraw-Hill/Irwin.

Maria Claudia Buzzi et al. (2010). "Accessing Google Docs via Screen Reader". In: *ICCHP 2010*. Ed. by K. Miesenberger et al. Springer-Verlag Berlin Heidelberg.

- Mark Lutz (2001). *Programming Python, 2nd Edition*. O'Reilly.
- Marty Alchin (2013). *Pro Django*. Apress.
- Médéric Morel et al. (2011). *Google Apps: Mastering Integration and Customization*. Packt Publishing Ltd.
- MySQL 5.5 Reference Manual (2013). url: <http://dev.mysql.com/doc/refman/5.5/en/index.html>. Diakses: 2013-10-07.
- OIT (2013). *Calendaring (online)*. url: <http://www.oit.umn.edu/google/using-google-calendar/index.htm>. Diakses: 2013-10-06.
- Refanidis, Ioannis and Anastasios Alexiadis. *SelfPlanner: An Intelligent Web-based Calendar Application*. University of Macedonia, Dept. of Applied Informatics, Thessaloniki, Greece.
- Stenchikova, Svetlana et al. (2007). "RavenCalendar: a Multimodal Dialog System for Managing a Personal Calendar". In: *in Proc. of HLT 2007*.
- UMS (2012). *Jadwal Terpadu (online)*. url: <https://ajt.ums.ac.id/>. diakses: 2013-10-06.