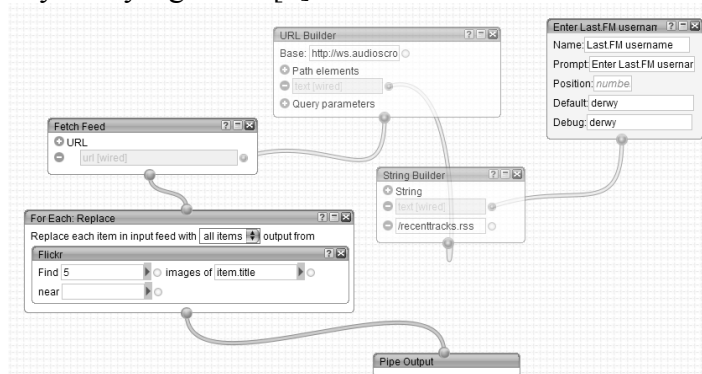


# 1. Pendahuluan

## 1.1 Latar belakang

Informasi yang tersedia di internet semakin bertambah dan meningkat dengan tajam. Informasi-informasi tersebut tersedia dalam berbagai format seperti teks dan multimedia.

Penyedia layanan informasi memberikan akses ke layanan mereka melalui API dalam berbagai format standar semisal RSS, ATOM, XML, JSON, REST yang menjadi standar pertukaran data yang populer digunakan. “*Mashup* adalah sebuah aplikasi web yang menggabungkan data dari satu atau lebih sumber menggunakan API standar tersebut” [7]. Darlene Fichter mengungkapkan “*mashup* adalah website atau aplikasi web yang menggunakan konten dari satu atau lebih sumber untuk membuat layanan yang baru” [4].



Gambar 1-1. Visualisasi mashup pada Yahoo! Pipe (<http://pipes.yahoo.com>)

Aplikasi situasional adalah perangkat lunak yang dibuat untuk sekelompok kecil pengguna dengan kebutuhan khusus dalam waktu yang sangat singkat[8]. Informasi dapat melalui proses *filtering* untuk menyesuaikan dengan kebutuhan dari suatu aplikasi situasional yang akan dibangun. Karakteristik aplikasi situasional adalah, cepat dikembangkan, mudah digunakan, tidak rumit dan melayani kebutuhan informasi spesifik. Aplikasi situasional dibangun dari mashup untuk mempersingkat fasa pengembangan perangkat lunak.

Namun pengguna *mashup* memerlukan pemahaman yang baik terhadap API dan operator yang digunakan, serta pemahaman terhadap alur informasi. Sehingga tidak sembarang user yang dapat menggunakan *mashup* untuk membangun aplikasi situasional.

Untuk mengatasi masalah tersebut, penggunaan wizard merupakan alternatif untuk mempermudah penggunaan *mashup*. Wizard adalah elemen dari antarmuka yang disajikan dalam kotak dialog-dialog yang akan menuntun user melakukan suatu pekerjaan dengan urutan tertentu[12]. Dalam tugas akhir ini akan melakukan implementasi dan analisis pengaruh penggunaan *wizard* dalam pembangunan aplikasi situasional pada sistem *mashup*.

## 1.2 Perumusan masalah

Hipotesis dari tugas akhir ini adalah:

- Wizard dapat diterapkan dalam sistem mashup.
- Wizard dapat mempermudah penggunaan sistem mashup dalam membangun aplikasi situasional konten berita.

Dari kesulitan yang ditemukan dalam penggunaan *mashup*, didapatkan beberapa permasalahan, diantaranya :

1. Bagaimana menerapkan *wizard* dalam sistem *mashup*?
2. Bagaimana pengaruh penggunaan *wizard* terhadap pembangunan aplikasi situasional konten berita?

Batasan masalah yang di definisikan dalam tugas akhir ini adalah :

1. *Mashup* yang dibangun adalah mashup dengan sumber data RSS dan ATOM. Menggunakan operator logika, dan operasi string sederhana.
2. Output dari mashup dalam format RSS.
3. Aplikasi yang akan dibangun adalah aplikasi berbasis web.

Tidak membahas masalah keamanan aplikasi, ketidaktersediaan data dan atau jaringan.

## 1.3 Tujuan

Tujuan pembuatan Tugas Akhir ini adalah :

1. Merancang dan membangun sistem *mashup* dengan fasilitas *wizard*
2. Menganalisis pengaruh penggunaan *wizard* dalam pembuatan aplikasi situasional dari *mashup* yang dibangun dengan melakukan usability testing terhadap *wizard*.

## 1.4 Metodologi penyelesaian masalah

Metodologi yang akan digunakan untuk menyelesaikan tugas akhir ini adalah :

1. Studi literatur.  
Menambah wawasan dari buku, jurnal-jurnal, artikel yang tersedia di Internet dan di sumber-sumber lainnya yang berhubungan dengan *mashup system*, *wizard*, *situational application* dan *usability testing*.
2. Perumusan masalah  
Mengidentifikasi masalah yang muncul dalam penggunaan *mashup* yang akan dibantu oleh *wizard*, yang hasilnya akan menjadi spesifikasi kebutuhan pada tahap analisis dan perancangan.
3. Analisis dan perancangan  
Melakukan analisis dan perancangan sistem *mashup*, *wizard* dan aplikasi situasional yang akan dibangun, meliputi fitur-fitur dalam sistem *mashup*, fitur-fitur *wizard* dan alur sistem. Sedangkan tahap perancangan meliputi

penyusunan *usecase*, basis data, antarmuka sistem, struktur aplikasi dan skenario pengujian.

4. Implementasi

Mengimplementasikan hasil dari tahap analisis dan perancangan. Yang dihasilkan dari tahap ini adalah *mashup editor* berbasis web dengan *wizard* sebagai alat bantu pembuatan aplikasi situasional untuk kebutuhan konten berita.

5. Pengujian Sistem

Tahap pengujian akan dilakukan dengan cara menggunakan *mashup* yang telah dibangun untuk membuat beberapa sampel aplikasi situasional untuk konten berita dengan menggunakan bantuan *wizard* dan secara manual (tanpa bantuan *wizard*). Selain itu, akan dilakukan *usability testing* terhadap *wizard* yang mencakup:

- Effectiveness : Performansi dalam menyelesaikan task.
- Learnability : Derajat pembelajaran dalam menyelesaikan task.
- Satisfaction: kepuasan user terhadap sistem, akan diukur dengan metode kuesioner.

6. Analisis Hasil Pengujian.

Pada tahap ini didapatkan hasil analisis dari *usability testing* dan analisis hasil kuesioner.

7. Penyusunan laporan Tugas Akhir dan membuat kesimpulan.

## 1.5 Sistematika Penulisan

Tugas akhir ini disusun berdasarkan sistematika sebagai berikut :

1. Pendahuluan

Dalam bab ini dimuat latar belakang masalah, perumusan dan batasan masalah, tujuan penelitian, metodologi penulisan dan sistematika penulisan.

2. Dasar Teori

Pada bab ini dimuat beberapa teori yang berkaitan dengan *mashup*, *wizard*, dan *usability testing*.

3. Analisis dan Perancangan Sistem

Pada bab ini dimuat tentang analisis dan pemodelan sistem secara keseluruhan.

4. Implementasi dan pengujian

Bab ini menjelaskan tahap implementasi sistem dan analisis dari setiap proses yang dijalankan.

5. Kesimpulan dan Saran

Pada bab ini dimuat tentang kesimpulan secara keseluruhan dan saran untuk pengembangan sistem selanjutnya.