

## PENDETEKSIAN CODE SMELL PADA WEBSITE PERUSAHAAN

Andi Jamiati Paramita, Andi Hutami Endang, Dian Aprilia Khairunnisa

Institut Teknologi dan Bisnis Kalla, Indonesia

Email: paramita@kallabs.ac.id, hutamiendang@kallabs.ac.id, dian@kallabs.ac.id

### Abstrak

Website merupakan pusat penyajian informasi yang penting dimiliki oleh perusahaan untuk dapat menjangkau konsumen lebih luas. Pengembangan website harus memperhatikan pemeliharaan dimulai dari penulisan kode. Oleh karena itu, penelitian ini mengembangkan sebuah platform untuk mendeteksi *code smell* pada file HTML website perusahaan/institusi dengan menggunakan metode Crowdsourcing. Hasil pendeteksian ini akan melewati fase voting untuk mendapatkan hasil yang lebih meyakinkan.

**Kata Kunci:** *website; code smell; crowdsourcing; voting*

### Abstract

*The website is the center of presentation of information that is important for the company to be able to reach consumers more broadly. Website development should pay attention to maintenance starting from writing code. Therefore, this study developed a platform to detect code smell on the HTML files of company websites/institutions using the Crowdsourcing method. The results of this detection will pass the voting phase to get a more convincing result.*

**Keywords:** *website; code smell; crowdsourcing; voting*

Received: 2021-12-20; Accepted: 2022-01-05; Published: 2022-01-15

### Pendahuluan

Tren digital yang dilakoni oleh beberapa perusahaan adalah mengembangkan website sebagai salah satu fitur kemudahan untuk penyebaran informasi. Website menjadi media perusahaan untuk menjangkau konsumen secara lebih luas (Yulianto, Dewi, & Wijaya, 2014). Jangkauan ini didukung dengan teknologi dan banyaknya pengguna internet. Mengingat hal itu maka kemampuan website untuk memenuhi kebutuhan calon konsumen perlu dijaga. Salah satu unsur penting dari website perusahaan adalah fase pemeliharaan dimana perlu dilakukan pendeteksian *code smell* agar pemeliharaan mudah dilakukan. Meskipun saat ini berbagai kakas otomatis untuk mendeteksi *code smell* pada website telah banyak berkembang, tetapi, keakuratannya masih perlu ditingkatkan dengan pendeteksian oleh manusia yang dibantu dengan sebuah platform.

Dalam bidang rekayasa perangkat lunak, kemudahan pemeliharaan (*maintenance*) sebuah perangkat lunak didukung kode-kode program yang ditulis dengan cara yang benar (Sommerville, 2011). Untuk memastikan hal tersebut, diperlukan pemeriksaan kembali (*code review*) untuk mencegah munculnya *code smell* dalam perangkat lunak. Berbagai teknik maupun platform telah dikembangkan untuk mendeteksi adanya *code smell* dalam kode program. Namun, beberapa penelitian yang mengembangkan alat deteksi otomatis untuk *code smell* terbatas pada beberapa kriteria, misalnya tingkat kompleksitas kode program, sehingga dibutuhkan pendeteksian secara manual oleh manusia untuk mengakuratkan hasil pendeteksian tersebut. Salah satu metode yang menggabungkan peran manusia dan komputasi mesin untuk menyelesaikan beberapa permasalahan pengembangan perangkat lunak adalah *crowdsourcing* (Mao, Capra, Harman, & Jia, 2017).

*Crowdsourcing* adalah suatu aktifitas atau tindakan yang dilakukan oleh suatu perusahaan atau institusi yang mengambil salah satu fungsi pekerjaan atau tugas yang seharusnya dilakukan oleh karyawannya menjadi disebarluaskan secara terbuka dan bebas untuk orang banyak atau kerumunan yang terkoneksi dengan jaringan komputer dalam hal ini Internet (Little, 2009). Dalam penelitian ini, kami mengimplementasikan platform pendeteksi *code smell* pada sistem informasi perusahaan berbasis web memanfaatkan *crowdsourcing* dengan metode kontrol kualitas menggunakan perhitungan *Vote*.

#### A. Code Smell

Code Smell adalah kode yang kehadirannya dapat menjadi masalah terutama bagi program yang memerlukan pemeliharaan (Fontana, Mäntylä, Zanoni, & Marino, 2016).

Secara teoritis, *code smell* merupakan implementasi penulisan kode yang kurang baik (Lenstra, 1994). Namun pada praktisnya, Code Smell berada diantara perancangan dan implementasi, dimana kehadirannya pada kode program untuk membantu perancangan kelas, namun secara konkrit tetap tertulis di dalam file (Khomh, Di Penta, & Guéhéneuc, 2009).

Pada awal definisinya, *code smell* terbagi menjadi 22 jenis. Namun, seiring waktu, kini *code smell* sudah dapat dikelompokkan berdasarkan jenis Bahasa pemrograman yang dibuat. Penelitian ini fokus pada *code smell* yang dapat dideteksi dari file kode HTML. Sonar Source sebagai salah satu perusahaan yang fokus pada pengukuran kualitas dan keamanan kode program memetakan 38 jenis *code smell* untuk file kode HTML berdasarkan rekomendasi yang telah dibuat oleh Wide World Web Consortium (W3C). Penelitian ini akan fokus pada 17 *code smell* yang dapat dilihat pada tabel 1 dibawah ini:

**Tabel 1**  
**Daftar Code Smell**

ID	Deskripsi	Tags
CS1	Kode yang menjadi komentar	Unused
CS2	Penggunaan “ <i>aria-label</i> ” atau “ <i>aria-labbelledby</i> ”	Accessibility
CS3	Penambahan <i>subtitle</i> pada video	Accessibility
CS4	Penggunaan tag “ <i>FIXME</i> ”	CWE
CS5	Tautan langsung pada gambar	Accessibility
CS6	Penggunaan tag “ <i>TO DO</i> ”	CWE
CS7	Penambahan copyright dan license pada header	-
CS8	Penggunaan atribut “ <i>Style</i> ”	-
CS9	Penggunaan elemen “ <i>Child</i> ”	-
CS10	Penggunaan atribut untuk elemen gambar	-
CS11	Kompleksitas isi file	-
CS12	Penambahan link langsung pada gambar (URIs)	-
CS13	Penambahan javascript pada file	Brain-overload
CS14	Penambahan ekspresi JSP	Jsp-jjsf obsolete
CS15	Penambahan ID untuk tiap elemen	convention
CS16	Penggunaan double quotes (“ ”) untuk setiap atribut	convention
CS17	Penggunaan tag heading	accessibility

- a. CS1: Code yang tidak digunakan sebaiknya dihapus dan tidak hanya ditandai dengan menjadikannya komentar. Code smell ini mengurangi kemampuan code untuk mudah dibaca.
- b. CS2: Atribut ‘Aria-label’ dan ‘Aria-labelled’ disertakan setiap kali penggunaan elemen seperti <nav> atau <aside>.
- c. CS3: Penambahan subtitle pada konten video membuat semua pengguna termasuk yang memiliki keterbatasan dapat mengerti informasi yang disampaikan.
- d. CS4: Hindari penggunaan tag *FIXME* untuk menandai bagian code yang perlu diperbaiki
- e. CS5: tautan yang melekat pada gambar sebaiknya langsung terarah pada halaman depan.
- f. CS6: Hindari penggunaan tag *TO DO* untuk menandai bagian code yang perlu ditambahkan
- g. CS7: Semua *file* harus diawali dengan *header* sebagai tanda kepemilikan *file* dan lisensi.
- h. CS8: Atribut style sebaiknya tidak disertakan dalam file html tetapi didefinisikan dalam file CSS yang terpisah.
- i. CS9: Penggunaan elemen anak dalam elemen induk. Contoh elemen induk adalah <head></head> dengan elemen child <title></title> yang harus berada dalam tag <head>.
- j. CS10: kelengkapan atribut untuk elemen gambar seperti *width*, *height* dan *alt*.

- k. CS11: penambahan tag untuk pengambilan keputusan (decision) seperti `&&` atau `||` akan menambah kompleksitas code.
- l. CS12: hindari menggunakan gambar dengan menautkan langsung dari website.
- m. CS13: hindari penulisan kode javascript pada file code html. Sebaiknya kode javascript dituliskan terpisah kedalam file .js
- n. CS14: hindari penulisan ekspresi JSP (menggunakan `<%= ... %>`)
- o. CS15: ID diperlukan untuk membantu memastikan bahwa semua elemen terkonfigurasi dengan ID yang berbeda-beda. ID dapat memudahkan pencarian elemen tertentu.
- p. CS16: sesuai ketentuan, penggunaan double quote (“ ”) lebih disarankan.
- q. CS17: Penggunaan tag `<h1>` dan seterusnya sebaiknya dituliskan berurut.

## **B. Crowdsourcing Software Engineering**

Crowdsourcing merupakan model penyelesaian masalah secara terdistribusi dengan mengkombinasikan komputasi mesin dengan peranan manusia (Budiarta, Ginting, & Simarmata, 2020). Istilah *Crowdsourced Software Engineering* (CSE) digunakan untuk penerapan crowdsourcing yang mendukung proses pengembangan perangkat lunak (Widiantika, Resika, & Santyadiputra, 2019). Mereka membangun definisi bahwa CSE -adalah Tindakan melakukan setiap tugas rekayasa perangkat lunak yang bersifat eksternal secara bebas oleh kelompok besar pekerja online yang tidak didefinisikan. Secara umum, CSE memiliki 3 aktor utama yaitu, requester yang memiliki tugas dibidang pengembangan perangkat lunak untuk diselesaikan, worker yang berpartisipasi menyelesaikan tugas dan platform sebagai perantara yang menyatukan requester dan *worker* (Mao et al., 2017). Salah satu platform crowdsourcing seperti Amazon Mechanical Turk (<https://mturk.com/mturk/>) mewadahi requester yang menawarkan task dan worker dapat mengaksesnya untuk menyelesaikan task tersebut. Platform ini juga menyediakan plug-in yang dapat digunakan oleh pihak lain untuk mempublikasikan *task* (Little, 2009). Umumnya, task yang diberikan telah melewati proses dekomposisi menjadi microtask sehingga worker tidak membutuhkan usaha besar dan waktu yang banyak untuk menyelesaikannya. Pada beberapa platform, worker juga diberikan imbalan atas hasil pengerjaannya. Penggunaan lain crowdsourcing juga diterapkan pada saat melakukan desain suatu sistem. Metode crowdsourcing dapat mengumpulkan banyak alternatif solusi, pengukuran kualitas solusi yang hadir, dan kemudahan penyelesaian design dengan berbagai tingkat kesulitan (Weidema, López, Nayebaziz, Spanghero, & Van Der Hoek, 2016).

## **Metode Penelitian**

### **A. Isu Penelitian**

Isu utama yang dibahas dalam penelitian ini adalah bagaimana *code smell* dapat diidentifikasi pada website perusahaan melalui metode *crowdsourcing* dengan bantuan platform yang dikembangkan. Pada penelitian ini, pendeteksian dilakukan pada halaman interface website dengan jenis HTML.

## B. Implementasi

Penelitian ini mengusung sistem yang dapat mendeteksi code smell pada file html website perusahaan dengan metode crowdsourcing. Metode ini menjadikan pengguna sebagai worker yang bertugas mendeteksi code smell dan hasil pendeteksian mereka akan dinilai kualitasnya berdasarkan perhitungan voting. Voting sendiri adalah perhitungan poin jika jawaban berupa code smell mampu dideteksi lebih dari 1(satu) orang worker.

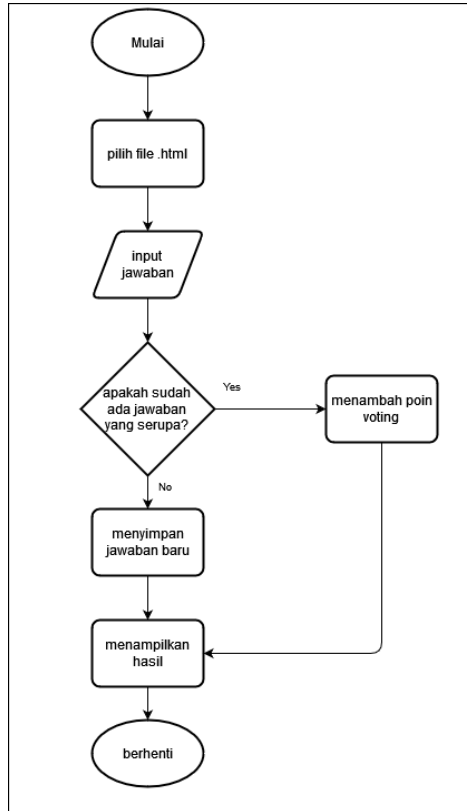
Implementasi dimulai dengan pengumpulan data berupa file html untuk tampilan website dari beberapa perusahaan lokal. Tabel 4.1 adalah daftar perusahaan yang telah diubah Namanya menjadi codename serta jumlah file html yang disertakan.

**Tabel 2**  
**Data Uji**

Codename	.html Files
AKR-01	1
DP-02	1
EM-03	1
FO-04	1
FPB-05	1
PLT-06	1
PWD-07	1
PWA-08	3
TPH-09	1

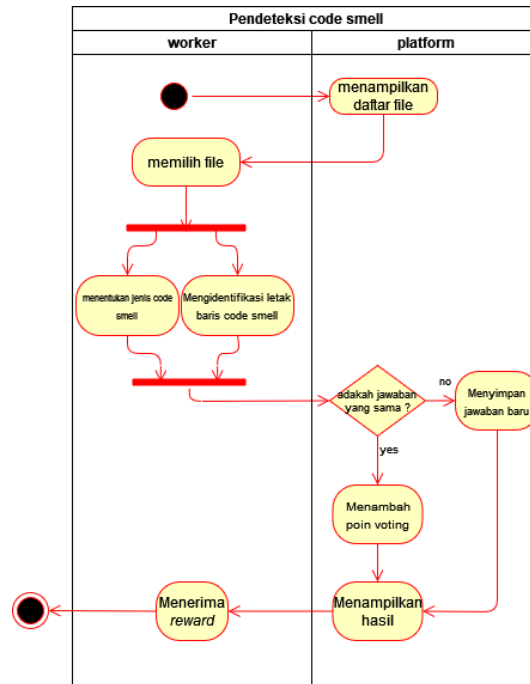
Penamaan codename mengambil akronim dari nama perusahaan atau instansi. Angka yang mengikuti codename berarti jumlah file yang menjadi data uji. Penamaan codename ini dibuat untuk mempermudah proses identifikasi code smell. Selain itu, worker tidak diperkenankan untuk mengetahui perusahaan atau instansi yang memiliki kode file demi menghindari keberpihakan worker saat memilih file

Selanjutnya tahap pendeteksian dilakukan dengan metode crowdsourcing yaitu sebuah file akan diperiksa oleh beberapa orang untuk mendapatkan *code smell*. Pada penelitian ini, orang yang mendeteksi code smell pada disebut sebagai worker. Para worker dapat melakukan pendeteksian pada lebih dari 1 file.



**Gambar 1**  
**Alur Pendeteksian *Code Smell***

Hasil dari code smell yang berhasil didapatkan akan melalui proses *voting* untuk menentukan keabsahannya. Gambar 1 menunjukkan alur pendeteksian code smell. Sementara itu, pembagian aksi antara worker dengan platform digambarkan pada gambar 2.



**Gambar 2**  
**Swimlane Activity Diagram**

Tahap terakhir adalah proses voting. Tahap ini berlangsung secara parallel dengan tahap pendeteksian code smell. Setiap jawaban worker berupa jenis code smell dan baris akan dihitung sebagai 1(satu) suara. Jika worker lain menemukan jawaban yang sama, maka code smell pada baris yang sama akan mendapatkan tambahan 1(satu) suara. Semakin banyak suara yang terkumpul menandakan jawaban tersebut menjadi prioritas maka tingkat keabsahannya akan semakin valid.

**Hasil dan Pembahasan**

Penelitian ini melibatkan 11 jumlah file html dan 3 jumlah worker untuk mendeteksi code smell pada website perusahaan. Hasil pendeteksian dijabarkan dalam tabel 3 berikut:

**Tabel 3**  
**Code Smell**

<b>Codename</b>	<b>Code smell terdeteksi</b>	<b>Jumlah worker</b>	<b>Poin voting</b>
AKR-01	3	3	3
DP-02	1	3	3
EM-03	3	3	2
FO-04	2	3	3
FPB-05	1	3	3
PLT-06	3	3	3
PWD-07	6	3	3
PWA-08	11	3	2
TPH-09	1	3	3

Jenis code smell yang paling sering dideteksi adalah “Penambahan copyright dan license pada header” (CS7). Code smell jenis ini muncul di semua file yang menjadi data uji pada penelitian ini. Selain itu, code smell dengan id CS1 yaitu “source code yang dijadikan komentar” menjadi code smell yang juga cukup banyak ditemukan yaitu pada 6 file.

Pada proses voting, seluruh hasil deteksi dari *worker* dihitung sebagai voting. Jika ada hasil deteksi yang sama, maka perhitungan voting bertambah 1. Dari hasil voting yang dilakukan terhadap data uji, terdapat 2 file yang memiliki nilai voting 2 (dua). Ini diartikan bahwa hampir semua jawaban yang diberikan setiap worker adalah sama.

### **Kesimpulan**

Berdasarkan hasil pengujian terhadap 11 file html yang dilakukan oleh 3 orang worker, dapat disimpulkan bahwa code smell pada tampilan website perusahaan dapat dideteksi dengan menggunakan metode crowdsourcing. Hasil pendeteksian diperkuat dengan metode voting sehingga keabsahan penemuan code smell ini lebih meyakinkan.

Untuk pengembangan penelitian selanjutnya diharapkan agar platform pendeteksi code smell ini dibuat menjadi plugin yang dapat ditautkan pada setiap pembangunan website di berbagai perusahaan.



## BIBLIOGRAFI

- Budiarta, K., Ginting, S. O., & Simarmata, J. (2020). *Ekonomi dan Bisnis Digital*. Yayasan Kita Menulis. [Google Scholar](#)
- Fontana, F. A., Mäntylä, M. V., Zanoni, M., & Marino, A. (2016). Comparing and experimenting machine learning techniques for code smell detection. *Empirical Software Engineering*, 21(3), 1143–1191. [Google Scholar](#)
- Khomh, F., Di Penta, M., & Guéhéneuc, Y. G. (2009). An exploratory study of the impact of code smells on software change-proneness. *Proceedings - Working Conference on Reverse Engineering, WCRE*, 75–84. <https://doi.org/10.1109/WCRE.2009.28>
- Lenstra, A. K. (1994). Factoring. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 857 LNCS, 28–38. <https://doi.org/10.1007/bfb0020422>
- Little, G. (2009). TurKit: Tools for iterative tasks on mechanical turk. *2009 IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2009*, (Figure 1), 252–253. <https://doi.org/10.1109/VLHCC.2009.5295247> [Google Scholar](#)
- Mao, K., Capra, L., Harman, M., & Jia, Y. (2017). A survey of the use of crowdsourcing in software engineering. *Journal of Systems and Software*, 126, 57–84. <https://doi.org/10.1016/j.jss.2016.09.015> [Google Scholar](#)
- Sommerville, I. (2011). *Software Engineering (9th ed.; Boston, Ed.)*. Massachusetts: Pearson Education. [Google Scholar](#)
- Weidema, E. R. Q., López, C., Nayebaziz, S., Spanghero, F., & Van Der Hoek, A. (2016). Toward Microtask crowdsourcing software design work. *Proceedings - 3rd International Workshop on CrowdSourcing in Software Engineering, CSI-SE 2016*, 41–44. <https://doi.org/10.1145/2897659.2897664> [Google Scholar](#)
- Widiantika, M., Resika, I. K., & Santyadiputra, G. S. (2019). God Temple (Geographic of Dewata Temple): Aplikasi Persebaran Pura Pada Perangkat Mobile Application Berbasis Crowdsourcing. *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, 8(2), 166–175. [Google Scholar](#)
- Yulianto, B., Dewi, L. C., & Wijaya, O. (2014). Peran Website Restoran terhadap Daya Tarik Konsumen Online. *ComTech: Computer, Mathematics and Engineering Applications*, 5(2), 1096. <https://doi.org/10.21512/comtech.v5i2.2376> [Google Scholar](#)

**Copyright holder:**

Andi Jamiati Paramita, Andi Hutami Endang, Dian Aprilia Khairunnisa (2022)

**First publication right:**

Syntax Literate: Jurnal Ilmiah Indonesia

**This article is licensed under:**

