

EVALUASI PHP STANDARDS RECOMMENDATIONS PADA PROYEK SHARIF JUDGE

NICHOLAS KHRISNA SANDYAWAN—6181801060

1 Data Skripsi

Pembimbing utama/tunggal: **Pascal Alfadian Nugroho**

Pembimbing pendamping: -

Kode Topik : **PAN5401**

Topik ini sudah dikerjakan selama : **1 semester**

Pengambilan pertama kali topik ini pada : Semester **54 - Genap 22/23**

Pengambilan pertama kali topik ini di kuliah : **Skripsi 1**

Tipe Laporan : **A** - Dokumen pendukung untuk **pengambilan ke-2 di Skripsi 1**

2 Latar Belakang

Pengembangan aplikasi berbasis web dengan bahasa PHP cukup populer di kalangan pengembang web. Bahkan tersedia banyak *framework* yang dapat digunakan untuk memudahkan pengembangannya. Walaupun demikian, masih cukup sering dijumpai masalah dalam pengembangan. Salah satunya adalah penulisan kode program yang tidak konsisten karena belum ditentukan aturan atau standar penulisan tertentu. Hal ini membuat suatu proyek aplikasi web menjadi rumit dan sulit dipelihara, terutama jika melibatkan banyak pengembang.

Salah satu proyek aplikasi berbasis web adalah SharIF Judge. SharIF Judge merupakan aplikasi berbasis web yang dapat digunakan untuk menilai kode program dalam bahasa C, C++, Java, dan Python. Aplikasi ini ditulis dengan *framework* PHP CodeIgniter dan bagian *backend* dibuat dengan BASH. SharIF Judge yang dibahas pada skripsi ini adalah *fork* dari Sharif Judge yang dibuat oleh Mohammad Javad Naderi. Versi *fork* ini sudah dikembangkan sesuai kebutuhan Teknik Informatika UNPAR untuk digunakan dalam proses penilaian pada beberapa mata kuliah.

PHP Standards Recommendation (PSR) adalah kumpulan standar penulisan PHP yang dibuat oleh PHP Framework Interop Group. Standar-standar ini berisi aturan dan rekomendasi penulisan kode program PHP yang dapat memudahkan pengembangan aplikasi agar lebih konsisten. Dokumentasinya dapat diakses di internet. Pada saat dokumen ini dibuat, terdapat 14 bab PSR yang berlaku untuk digunakan (Accepted) antara lain:

- PSR-01: Basic Coding Standard
- PSR-03: Logger Interface
- PSR-04: Autoloading Standard
- PSR-06: Caching Interface
- PSR-07: HTTP Message Interface
- PSR-11: Container Interface
- PSR-12: Extended Coding Style Guide
- PSR-13: Hypermedia Links

- PSR-14: Event Dispatcher
- PSR-15: HTTP Handlers
- PSR-16: Simple Cache
- PSR-17: HTTP Factories
- PSR-18: HTTP Client
- PSR-20: Clock

Standar yang digunakan hanya dari bab-bab yang berstatus “Accepted”. Bab dengan status lain yaitu “Draft”, “Abandoned”, dan “Deprecated” tidak akan digunakan sehingga tidak dicantumkan dalam dokumen skripsi.

Pada skripsi ini, kode-kode program PHP pada SharIF Judge akan diperiksa secara umum dan dievaluasi seberapa jauh standar PSR yang sudah dipenuhi. Selanjutnya akan dibuat rekomendasi untuk meningkatkan jumlah standar PSR yang dapat dipenuhi berdasarkan hasil evaluasi tersebut. Masih akan ditentukan lagi strategi untuk melakukannya, misalnya menggunakan bantuan *tools* tertentu untuk automasi, bab-bab PSR apa saja yang relevan untuk digunakan sebagai standar evaluasi, dan seberapa banyak kode yang harus diperiksa secara manual.

3 Rumusan Masalah

- Seberapa jauh PSR yang sudah terpenuhi ada SharIF Judge?
- Bagaimana mengevaluasi kode PHP pada SharIF Judge sesuai PSR?
- Bagaimana memberikan rekomendasi perbaikan pada kode PHP SharIF Judge agar meningkatkan jumlah PSR yang terpenuhi?

4 Tujuan

Tujuan yang ingin dicapai dalam penulisan skripsi ini sebagai berikut:

- Mengetahui seberapa jauh PSR yang sudah terpenuhi pada SharIF Judge.
- Mengevaluasi kode PHP pada SharIF Judge sesuai PSR.
- Memberikan rekomendasi perbaikan pada kode PHP SharIF Judge agar meningkatkan jumlah PSR yang terpenuhi.

5 Detail Perkembangan Pengerjaan Skripsi

Detail bagian pekerjaan skripsi sesuai dengan rencana kerja/laporan perkembangan terakhir :

1. Mempelajari SharIF Judge versi *fork*.

Status : Ada sejak rencana kerja skripsi.

Hasil : Studi tentang SharIF Judge belum selesai dan masih dipelajari lebih lanjut. Isinya dalam Landasan Teori masih sebagian yang selesai. SharIF Judge menggunakan *framework* CodeIgniter dengan struktur yang baru bagi saya. Target kode program PHP yang akan dievaluasi adalah bagian yang bukan dari *template* CodeIgniter.

2. Melakukan studi literatur mengenai PSR dan PHP *linter*.

Status : Ada sejak rencana kerja skripsi.

Hasil : Berikut adalah perincian dari hasilnya.

- Penulisan standar-standar PSR ini diterjemahkan dari bahasa Inggris ke bahasa Indonesia. Salah satu tantangan dalam menerjemahkan PSR ini adalah tidak setiap aturan ditulis dalam bentuk *list* yang mudah untuk dibaca. Banyak di antaranya merupakan bagian dari suatu paragraf yang berkelanjutan dengan konteksnya masing-masing sehingga penjelasan atau deskripsinya perlu dicantumkan pula dalam Landasan Teori agar tidak ada standar yang bermakna ambigu atau tidak jelas. Untuk saat ini, PSR dicantumkan semuanya dalam Landasan Teori kecuali dua bab yaitu PSR-07 dan PSR-12. Hal ini dikarenakan jumlah aturan yang harus diterjemahkan dan ditulis kembali cukup banyak sehingga dikhawatirkan akan memakan waktu yang lama. Untuk sementara, hal ini dicantumkan dalam Batasan Masalah. Jika memungkinkan, kedua bab ini akan disertakan semuanya dalam Landasan Teori nantinya.
- Ada beberapa kata kunci yang harus ditaati dalam standar PSR berdasarkan dokumen Request For Comments (RFC) 2119, yaitu “MUST”, “MUST NOT”, “SHOULD”, “SHOULD NOT”, dan “MAY”. Setiap kata kunci ditulis dengan huruf kapital dan memiliki tingkat prioritas yang berbeda dalam hal keharusannya. Selain itu ada beberapa sinonim dari kata kunci yang dipakai juga dalam PSR dengan huruf kapital, misalnya “REQUIRED” berada di tingkat yang sama dengan “MUST” yang berarti aturan dalam standar tersebut harus dipenuhi. Maka dari itu, setiap kata kunci dikelompokkan berdasarkan tingkatan tersebut dan diterjemahkan ke bahasa Indonesia agar penulisan dokumen skripsi menjadi konsisten dan tidak mengubah makna dari standar yang asli.
- Setiap standar diberikan kode tertentu agar mudah untuk dirujuk nantinya dari bab-bab lain dalam dokumen skripsi. Kode tersebut ditulis dengan format PSR-XXYY, di mana XX adalah nomor bab dan YY adalah nomor standar dalam bab tersebut. Sebagai contoh PSR-0402 berarti mengacu pada standar PSR nomor 2 dari bab PSR-04.
- PHP *linter* pada dasarnya digunakan untuk membantu melakukan pengecekan penulisan kode PHP secara keseluruhan dengan standar tertentu. Saya sudah mencoba PHP *linter* yang dibuat oleh pengguna GitHub dengan nama Brueggern. *Linter* ini hanya mengecek berdasarkan PSR-02 (Deprecated) dan PSR-12. Fiturnya dapat menunjukkan kesalahan penulisan dan memberikan rekomendasi perbaikannya. Satu fitur lainnya yaitu perintah untuk memperbaiki kesalahan-kesalahan tersebut. Namun setelah dipelajari lebih lanjut, PHP *linter* tersebut pada dasarnya menggunakan PHP Coding Standards Fixer (PHP CS Fixer) yang dimodifikasi. Pada dokumentasinya, PHP CS Fixer mengacu pada PSR-01, PSR-02 (Deprecated), dan PSR-12. Maka dari itu, masih akan dipelajari lebih lanjut tentang PHP CS Fixer dan ditentukan apakah hanya PHP CS Fixer yang dibutuhkan atau perlu dicantumkan juga PHP *linter* Brueggern dalam Landasan Teori.

3. Mengevaluasi PHP dari SharIF Judge sesuai dengan PSR.

Status : Ada sejak rencana kerja skripsi.

Hasil : Proses evaluasi belum dilakukan karena belum ditentukan analisis dan rencana untuk melakukannya. Sebelum itu, harus dilakukan studi literatur SharIF Judge yang lengkap untuk menentukan proses evaluasinya seperti:

- banyaknya kode PHP bukan *template* yang akan diperiksa, baik secara otomatis dengan bantuan *linter* maupun manual satu per satu sesuai standar PSR yang relevan
- perkiraan waktu yang dibutuhkan untuk evaluasi
- menentukan pengerjaan evaluasi untuk sebagian atau seluruh kode PHP yang sesuai jika memungkinkan

4. **Menguji SharIF Judge yang sudah dievaluasi.**

Status : Ada sejak rencana kerja skripsi.

Hasil : Belum terlaksana karena proses evaluasi belum dilakukan.

5. **Memberikan rekomendasi sesuai hasil evaluasi.**

Status : Ada sejak rencana kerja skripsi.

Hasil : Belum terlaksana karena SharIF Judge belum dievaluasi.

6. **Menulis dokumen skripsi.**

Status : Ada sejak rencana kerja skripsi.

Hasil : Dokumen skripsi yang sudah dibuat adalah Bab 1 Pendahuluan dan sebagian besar Bab 2 Landasan Teori, terutama PSR.

6 Pencapaian Rencana Kerja

Langkah-langkah kerja yang berhasil diselesaikan dalam Skripsi 1 ini adalah sebagai berikut:

1. Membuat dokumen skripsi Bab 1 Pendahuluan dan sebagian besar Bab 2 Landasan Teori
2. Melakukan studi literasi PSR
3. Mencoba salah satu *linter* untuk melakukan automasi pemeriksaan *file* PHP pada SharIF Judge yang berdasar pada PSR-01 dan PSR-12

7 Kendala yang Dihadapi

Kendala - kendala yang dihadapi selama mengerjakan skripsi :

- Terlambat mengumpulkan dokumen Rencana Kerja Skripsi sehingga tidak dapat melakukan Review Skripsi 1
- Adanya penundaan kerja atau prokrastinasi dalam pembuatan skripsi
- Ada beberapa masalah pribadi yang cukup menyita waktu dan konsentrasi, meliputi masalah keluarga, masalah diri sendiri, dan sakit yang berdurasi cukup lama.

Bandung, 19/06/2023

Nicholas Khrisna Sandyawan

Menyetujui,

Nama: Pascal Alfadian Nugroho
Pembimbing Tunggal