

SKRIPSI

IMPLEMENTASI EDITOR KODE PADA SHARIF JUDGE



Nicholas Aditya Halim

NPM: 2017730018

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2022**

UNDERGRADUATE THESIS

CODE EDITOR IMPLEMENTATION ON SHARIF JUDGE



Nicholas Aditya Halim

NPM: 2017730018

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2022**

LEMBAR PENGESAHAN

IMPLEMENTASI EDITOR KODE PADA SHARIF JUDGE

Nicholas Aditya Halim

NPM: 2017730018

Bandung, 23 Juni 2022

Menyetujui,

Pembimbing

Digitally signed
by Pascal
Alfadian Nugroho

Pascal Alfadian, Nugroho, M.Comp.

Ketua Tim Penguji

Digitally signed
by Raymond
Chandra Putra

Raymond Chandra Putra, M.T.

Anggota Tim Penguji

Digitally signed
by Elisati Hulu

Elisati Hulu, M.T.

Mengetahui,

Ketua Program Studi

Digitally signed
by Mariskha Tri
Adithia

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

IMPLEMENTASI EDITOR KODE PADA SHARIF JUDGE

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 23 Juni 2022



Nicholas Aditya Halim
NPM: 2017730018

ABSTRAK

Situasi pandemi Covid-19 menyebabkan seluruh kegiatan kuliah dilaksanakan secara *online*. Pada umumnya, kegiatan praktikum dan ujian pada mata kuliah pemrograman Teknik Informatika Unpar dapat diawasi secara langsung oleh dosen dan asisten dosen di lab komputer. Namun, pengawasan menjadi lebih sulit untuk dilakukan saat kuliah dilaksanakan secara *online*.

Pada mata kuliah pemrograman Teknik Informatika Unpar, dimanfaatkan SharIF Judge, sebuah *online judge* (sebuah sistem *online* yang berfungsi untuk mengevaluasi kode program) untuk mempermudah proses pengumpulan dan penilaian kode program. Pada SharIF judge, pengumpulan kode dilakukan dengan mengunggah kode program, yang biasanya dibuat oleh mahasiswa dengan menggunakan aplikasi eksternal. Karena proses pembuatan kode program tidak terintegrasi, SharIF Judge tidak memiliki kemampuan untuk mengawasi proses pembuatan kode program.

Pada skripsi ini, diimplementasikan *Integrated Development Environment* (IDE), sebuah aplikasi dengan kemampuan untuk mengedit, mengompilasi, dan menjalankan kode program [1] pada SharIF Judge, sehingga seluruh proses pembuatan kode program dapat dilakukan langsung di dalam SharIF Judge. Dengan implementasi IDE, seluruh proses pembuatan kode program terintegrasi pada SharIF Judge, dan selanjutnya dapat dimanfaatkan untuk pengawasan.

SharIF Judge dengan implementasi IDE diuji selama mata kuliah Dasar-dasar Pemrograman semester ganjil 2021/2022 Teknik Informatika Unpar. Berdasarkan hasil pengujian, seluruh masalah yang ditemukan berhasil diperbaiki, dan seluruh fitur yang diimplementasikan berfungsi dengan baik.

Kata-kata kunci: *Online judge, Integrated Development Environment*

ABSTRACT

The Covid-19 Pandemic caused every learning activities to be done online. Usually, practical lectures and exams in Unpar Informatics Engineering can be supervised directly by lecturers and assistants in the computer lab. However, supervision becomes more difficult to do when lectures are carried out online.

SharIF Judge is an online judge (an online system that serves to evaluate program code) utilized on several programming courses in Unpar Informatics Engineering Study Program to help with code submission and scoring. Code submission in SharIF Judge was done by file upload, which are made by students using external applications. SharIF Judge was unable to supervise code development process they are not integrated within SharIF Judge.

By implementing Integrated Development Environment (IDE), an application with the ability to write, compile, and run code on SharIF Judge, all code development process is now integrated, and further features can be added to help supervise students during learning activities.

SharIF Judge with IDE implemented is tested on odd semester 2021/2022 *Dasar-dasar Pemrograman* (Basic Programming) course in Unpar Informatics Engineering. Based on the results, every problem encountered has been fixed successfully and all implemented features have performed adequately.

Keywords: Online judge, Integrated Development Environment

*Dipersembahkan untuk Informatika UNPAR, keluarga,
teman-teman, dan diri sendiri*

KATA PENGANTAR

Puji syukur kepada Tuhan yang Maha Esa atas berkat-Nya dalam menyelesaikan skripsi yang berjudul **Implementasi Editor Kode pada SharIF Judge** ini. Penulis juga mengucapkan terima kasih kepada berbagai pihak yang telah mendukung penyusunan skripsi ini:

1. Keluarga yang selalu mendukung dan mendoakan penulis.
2. Bapak Pascal Alfadian, Nugroho, M.Comp. yang telah membimbing dan mendukung penulis dalam menyelesaikan skripsi.
3. Bapak Raymond Chandra Putra, M.T. dan Bapak Elisati Hulu, M.T. yang telah menguji dan memberi masukan untuk skripsi ini.
4. Teman-teman yang telah memberi bantuan dan dukungan selama penyusunan skripsi ini.
5. Pihak-pihak lain yang terlibat dalam penyusunan skripsi ini.

Penulis memohon maaf apabila terdapat kekurangan dan kesalahan pada skripsi ini. Semoga skripsi ini dapat memberi manfaat bagi yang membutuhkan.

Bandung, Juni 2022

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	2
1.6 Sistematika Pembahasan	3
2 LANDASAN TEORI	5
2.1 SharIF Judge	5
2.1.1 Instalasi	5
2.1.2 Fitur-fitur	5
2.2 CodeIgniter 3	6
2.2.1 Model-View-Controller	7
2.2.2 URL CodeIgniter	8
2.3 Twig	8
2.4 Program-program Linux	9
2.4.1 Bash	9
2.4.2 grep	9
2.4.3 sed	10
2.5 Integrated Development Environment	11
2.6 PDF.js	11
2.7 Ace	11
3 ANALISIS	15
3.1 Analisis Sistem Kini	15
3.1.1 Istilah-istilah Umum	15
3.1.2 Halaman-halaman pada SharIF Judge	16
3.1.3 Model, View, Controller	22
3.1.4 Penyimpanan Kode	31
3.1.5 Antrean Penilaian Kode	31
3.2 Analisis Sistem Usulan	32
3.2.1 Fitur-fitur	32
3.2.2 Use Case	34
3.2.3 Sequence Diagram	35
4 PERANCANGAN	37

4.1	Rancangan Antarmuka	37
4.2	Rancangan Perubahan Kode	38
4.2.1	Menampilkan soal	38
4.2.2	Mengedit Kode	38
4.2.3	Menyimpan Kode	39
4.2.4	Memuat Kode	39
4.2.5	Menjalankan Kode dengan Tes Kasus	39
4.2.6	Mengumpulkan Kode Melalui IDE	40
5	IMPLEMENTASI DAN PENGUJIAN	41
5.1	Lingkungan Implementasi dan Pengujian	41
5.2	Implementasi	42
5.2.1	Tampilan Antarmuka	42
5.2.2	Menampilkan Soal	42
5.2.3	Editor Kode	43
5.2.4	Menyimpan dan Memuat Kode	43
5.2.5	Menjalankan Kode dengan Tes Kasus	43
5.2.6	Mengumpulkan Kode Melalui IDE	45
5.3	Pengujian Fungsional	45
5.4	Pengujian Eksperimental	46
5.4.1	Perubahan melalui GitHub	46
5.4.2	Survei	50
6	KESIMPULAN DAN SARAN	61
6.1	Kesimpulan	61
6.2	Saran	61
	DAFTAR REFERENSI	63
	A KODE PROGRAM CONTROLLER	65
	B KODE PROGRAM MODEL	69
	C KODE PROGRAM VIEW	71
	D KODE PROGRAM LAINNYA	75

DAFTAR GAMBAR

1.1 Tahapan penelitian SharIF Judge	2
2.1 Use Case Diagram SharIF Judge	6
2.2 <i>Flow Chart</i> CodeIgniter	6
3.1 Bagan hubungan <i>assignment</i> , <i>problem</i> , dan <i>submission</i> pada SharIF Judge	16
3.2 Halaman Dashboard	16
3.3 Halaman Settings	17
3.4 Halaman Users	17
3.5 Halaman Notifications	18
3.6 Halaman Assignments	18
3.7 Halaman Problems	19
3.8 Halaman Submit	19
3.9 Halaman Final Submissions	20
3.10 Halaman All Submissions	20
3.11 Halaman Scoreboard	21
3.12 Halaman Hall of Fame	21
3.13 Halaman 24-Hour Log	22
3.14 Diagram kelas SharIF Judge	23
3.15 Tampilan Dashboard pada <code>dashboard.twig</code>	27
3.16 Tampilan <i>top bar</i> pada <code>top_bar.twig</code> dan <i>side bar</i> pada <code>side_bar.twig</code>	27
3.17 Tampilan halaman <i>error</i> 404 pada <code>error_404.php</code>	28
3.18 Use Case Diagram Fitur Usulan	34
3.19 Sequence Diagram Menjalankan Kode dengan Tes Kasus	36
4.1 Rancangan antarmuka halaman Submit	37
4.2 Diagram kelas perubahan pada SharIF Judge ¹	38
5.1 Tampilan antarmuka halaman Submit	42
5.2 Tampilan antarmuka setelah perubahan	47
5.3 Tampilan antarmuka unggah <i>file</i>	47
5.4 Tampilan antarmuka IDE	48
5.5 Hasil survei bagian 1 pertanyaan 1	51
5.6 Hasil survei bagian 1 pertanyaan 2	52
5.7 Hasil survei bagian 1 pertanyaan 4	52
5.8 Hasil survei bagian 2 pertanyaan 1	53
5.9 Hasil survei bagian 2 pertanyaan 2	53
5.10 Hasil survei bagian 2 pertanyaan 4	54
5.11 Hasil survei bagian 3 pertanyaan 1	54
5.12 Hasil survei bagian 3 pertanyaan 2	55
5.13 Hasil survei bagian 3 pertanyaan 4	55
5.14 Hasil survei bagian 4 pertanyaan 1	56
5.15 Hasil survei bagian 4 pertanyaan 2	56

5.16 Hasil survei bagian 4 pertanyaan 4	57
5.17 Hasil survei bagian 5 pertanyaan 1	57
5.18 Hasil survei bagian 5 pertanyaan 2	58
5.19 Hasil survei bagian 5 pertanyaan 4	58

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Dengan adanya situasi pandemi Covid-19 mulai tahun 2020, seluruh kegiatan kuliah wajib dilaksanakan secara *online*. Sebelum pandemi, kegiatan praktikum dan ujian kuliah pemrograman Teknik Informatika Unpar dilaksanakan di lab komputer, sehingga dapat diawasi secara langsung oleh dosen dan asisten dosen. Namun, pengawasan menjadi lebih sulit untuk dilakukan saat kuliah dilaksanakan secara *online*.

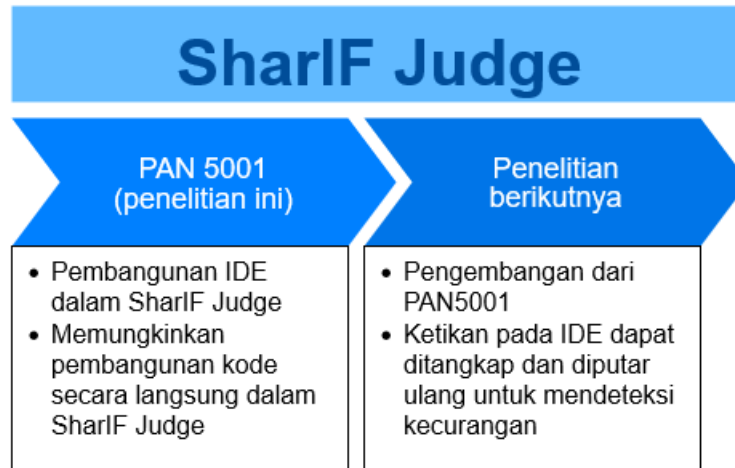
Pada mata kuliah pemrograman Teknik Informatika Unpar, dimanfaatkan SharIF Judge, sebuah *online judge* untuk bahasa pemrograman C, C++, Java dan Python untuk mempermudah proses pengumpulan dan penilaian kode program. *Online judge* adalah sebuah sistem *online* yang berfungsi untuk mengevaluasi kode program yang dikumpulkan oleh pengguna. Kode program yang dikumpulkan akan dikompilasi dan diuji pada lingkungan yang serupa, kemudian diberi nilai berdasarkan hasil yang didapatkan [2].

SharIF Judge (dengan IF kapital) adalah modifikasi cabang oleh Stillmen Vallian terhadap Sharif Judge (dengan if kecil) yang diciptakan oleh Mohammad Javad Naderi dengan *framework* CodeIgniter dan Bash [3]. Modifikasi ini dilakukan untuk menyesuaikan Sharif Judge terhadap kebutuhan spesifik Teknik Informatika Unpar [4].

Pada SharIF judge, pengumpulan kode dilakukan dengan mengunggah kode program, yang biasanya dibuat oleh mahasiswa dengan menggunakan aplikasi eksternal. Karena proses pembuatan kode program tidak terintegrasi, SharIF Judge tidak memiliki kemampuan untuk mengawasi proses pembuatan kode program. Sebelum pandemi, pengawasan dapat dilakukan secara langsung oleh dosen dan asisten dosen di lab komputer. Namun, ketika kuliah dilakukan secara *online*, diperlukan cara lain untuk mengawasi mahasiswa.

Pada skripsi ini, diimplementasikan *Integrated Development Environment* (IDE), sebuah aplikasi dengan kemampuan untuk mengedit, mengompilasi, dan menjalankan kode program pada SharIF Judge, sehingga seluruh proses pembuatan kode program dapat dilakukan langsung di dalam SharIF Judge. Dengan demikian, proses pembuatan kode terintegrasi pada SharIF Judge, dan dapat dimanfaatkan untuk pengawasan.

Gambar 1.1 menggambarkan tahapan penelitian SharIF Judge. Pada skripsi ini, diimplementasikan IDE pada SharIF Judge, sehingga seluruh proses pembuatan kode program dapat dilakukan langsung di dalam SharIF Judge. Pada penelitian selanjutnya, dapat ditambahkan fitur pada IDE untuk membantu pengawasan terhadap mahasiswa selama kegiatan kuliah, misalnya dengan menangkap dan memutar ulang ketikan pada IDE.



Gambar 1.1: Tahapan penelitian SharIF Judge

1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas pada skripsi ini adalah sebagai berikut:

1. Bagaimana mengimplementasikan *Integrated Development Environment* sehingga mahasiswa dapat mengedit, mengompilasi, dan menjalankan kode dalam SharIF Judge?
2. Bagaimana tanggapan pengguna terhadap implementasi *Integrated Development Environment* pada SharIF Judge?

1.3 Tujuan

Tujuan yang ingin dicapai skripsi ini adalah sebagai berikut:

1. Mengimplementasikan *Integrated Development Environment* sehingga mahasiswa dapat mengedit, mengompilasi, dan menjalankan kode dalam SharIF Judge.
2. Mendapatkan umpan balik dari tanggapan pengguna terhadap implementasi *Integrated Development Environment* pada SharIF Judge.

1.4 Batasan Masalah

Pada pengerjaan skripsi ini terdapat batasan sebagai berikut:

- Perangkat lunak SharIF Judge hanya digunakan pada lingkungan Teknik Informatika Unpar.
- Perangkat lunak hanya dapat diuji pada mata kuliah pemrograman di mana dosen pembimbing terlibat.
- Pada mata kuliah pemrograman, digunakan 2 *judge* terpisah untuk latihan dan kuis. Berdasarkan keputusan dosen koordinator, perangkat lunak hanya akan diuji pada *judge* latihan.

1.5 Metodologi

Metodologi pengerjaan skripsi ini adalah sebagai berikut:

1. Melakukan studi mengenai komponen yang diperlukan untuk membuat IDE berbasis web.
2. Mempelajari struktur SharIF Judge.
3. Merancang IDE berbasis web untuk SharIF Judge.
4. Mengimplementasikan IDE berbasis web pada SharIF Judge.
5. Melakukan pengujian dan eksperimen.
6. Menulis dokumen skripsi.

1.6 Sistematika Pembahasan

Sistematika pembahasan skripsi ini adalah sebagai berikut:

- Bab 1 Pendahuluan
Membahas latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan.
- Bab 2 Landasan Teori
Membahas teori-teori yang berhubungan dengan penelitian ini, yaitu SharIF Judge, CodeIgniter 3, Twig, Bash, Integrated Development Environment, PDF.js, dan Ace.
- Bab 3 Analisis
Membahas analisis terhadap perangkat lunak SharIF Judge.
- Bab 4 Perancangan
Membahas perancangan fitur yang diimplementasikan pada SharIF Judge.
- Bab 5 Implementasi dan Pengujian
Membahas implementasi fitur pada SharIF Judge dan pengujian yang dilakukan.
- Bab 6 Kesimpulan dan Saran
Membahas kesimpulan dari penelitian ini dan saran untuk penelitian berikutnya.

