

Sistem *StressLevel* Untuk Mengukur Tingkat Stres Pada Data Mahasiswa di Masa Pandemi

Ester Manda Caroline¹, Iin Ernawati², Ika Nurlaili Isnainiyah³,
S1 Sistem Informasi / Fakultas Ilmu Komputer

Universitas Pembangunan Nasional “Veteran” Jakarta

Jl. RS. Fatmawati Raya, Pd. Labu, Kec. Cilandak, Jakarta Selatan, DKI Jakarta, 12450, Indonesia

estermc@upnvj.ac.id¹, iin_ernawati@yahoo.com², nurlailika@upnv.ac.id³

Abstrak. Dalam menekan angka penyebaran COVID-19 di Indonesia yang merangkak naik, pemerintah Indonesia memutuskan untuk menerapkan sistem *social distancing* seperti memberlakukan sistem belajar jarak jauh melalui sistem daring. Stres yang meningkat di kalangan pelajar, khususnya mahasiswa, merupakan isu baru yang ditimbulkan oleh diberlakukannya sistem pembelajaran jarak jauh selama pandemi dari rumah. Diperlukan suatu metode untuk memantau tingkat stres agar dapat lebih mengetahui tingkat stres seseorang dan cara mengatasinya. Alat ukur DASS 42 parameter stres ini memberikan kemudahan untuk mendapatkan informasi tentang stres serta solusi yang tepat dengan memanfaatkan teknologi informasi. Metode *waterfall* digunakan dalam pengembangan sistem, sedangkan metode *blackbox testing* digunakan dalam pengujian sistem yang dibangun yaitu sistem *StressLevel*. Berdasarkan *blackbox testing* yang telah dilakukan terhadap sistem yang dibuat yaitu sistem *StressLevel* menunjukkan bahwa seluruh fitur yang ada pada sistem sudah mendapatkan status berhasil, sehingga sistem *StressLevel* sudah berfungsi sesuai dengan yang diharapkan.

Kata Kunci: tingkat stres, DASS 42, sistem, *StressLevel*

1 Pendahuluan

World Health Organization (WHO) mengukuhkan *Corona Virus Disease (COVID-19)* sebagai pandemi pada tanggal 11 Maret 2021 (Kementerian Kesehatan RI, 2020). Usaha untuk menekan angka penyebaran COVID-19 di Indonesia yang merangkak naik, pemerintah Indonesia memutuskan untuk menerapkan sistem *social distancing* seperti memberlakukan sistem belajar jarak jauh melalui sistem daring mulai dari tingkat dasar hingga perguruan tinggi [1]. Berlakunya sistem belajar jarak jauh selama masa pandemi dari rumah memicu suatu permasalahan baru, yaitu peningkatan stres pada peserta didik, terutama pada mahasiswa. Berdasarkan hasil survei tentang kesehatan mental selama masa pandemi yang dilakukan oleh Perhimpunan Dokter Spesialis Kedokteran Jiwa Indonesia (PDSKJI), menunjukkan bahwa sebesar 64.8% masyarakat mengalami masalah psikologi dengan golongan usia terbanyak pada 17-29 tahun dan diatas 60 tahun [2].

Dalam dunia psikologi, terdapat salah satu tes kepribadian yang digunakan sebagai pengukur kesehatan mental pada seseorang. Tes kepribadian tersebut menggunakan sebuah alat ukur yang disebut DASS 42 yang terdiri dari 42 pertanyaan untuk tiga skala yaitu depresi, kecemasan dan stres. Hasil dari tes kepribadian ini akan memberikan informasi mengenai tingkat dari masing-masing skala yang dialami, sehingga dapat dimanfaatkan untuk membantu dalam pengendalian tingkat kesehatan mental terutama stres dari seseorang [3].

Pemanfaatan teknologi informasi dalam membuat sistem dapat membantu mahasiswa lebih mudah untuk mengukur tingkat stres yang sedang dialami. Penggunaan teknologi di era ini dapat digunakan dalam mengukur tingkat stres secara efisien. Berdasarkan penelitian yang dilakukan oleh Syafitri 2019 [4] menunjukkan bahwa web responsif bernama WARAS dapat membantu pengguna untuk lebih peduli terhadap kesehatan mental yang mungkin sedang mereka alami dan dengan cepat menentukan kondisi kesehatan mental mereka. Berdasarkan penelitian ini, penulis memutuskan untuk merancang sistem pengukuran tingkat stres pada data mahasiswa pada masa pandemi dengan menggunakan parameter stres dari alat ukur DASS 42 untuk memudahkan mahasiswa dalam mempelajari tingkat stres yang sedang dialami serta mendapatkan informasi untuk mengatasi tingkat stres tersebut dengan memanfaatkan kemajuan teknologi pada saat ini.

2 Tinjauan Pustaka

2.1 Stres

Perasaan yang dapat dialami oleh setiap orang saat berusaha untuk mengatasi hambatan yang berkaitan dengan pendapatan, pekerjaan, hubungan, lingkungan, pendidikan dan keadaan lainnya disebut sebagai stres [5]. Stres adalah sebuah reaksi fisik dan psikologis dari rasa tertekan atau tidak nyaman yang dialami oleh seseorang terhadap suatu kondisi atau insiden baik internal maupun eksternal.

2.1.1 Tingkat Stres

Tingkat stres pada manusia dibagi menjadi tiga, yaitu stres ringan, stres sedang dan stres berat. Stres ringan merupakan tingkatan stres yang sering dirasakan oleh semua orang dan biasanya tidak memiliki efek fisiologis yang negatif, sedangkan stres sedang dan berat dapat menimbulkan suatu penyakit. Stres ringan biasanya akan hilang dalam hitungan menit atau jam, lalu untuk stres sedang dapat terjadi lebih lama dalam hitungan jam hingga hari. Sedangkan stres berat dianggap kronis karena dapat terjadi dalam hitungan minggu ataupun tahun [6].

2.1.2 DASS 42

Banyak instrumen penilaian psikologis, terutama yang berhubungan dengan tes kepribadian telah dibuat hingga saat ini. *Depression Anxiety Stress* (DASS), yang dibuat oleh Lovibond dan Lovibond pada tahun 1995, merupakan salah satu instrumen penilaian psikologis tersebut. Instrumen penilaian psikologis DASS yang terdiri dari 42 item digunakan untuk menilai tekanan psikologis secara keseluruhan mulai dari depresi, kecemasan, hingga stres. Tes ini terdiri dari tiga skala, masing-masing memiliki 14 item yang kemudian dipecah menjadi sejumlah subskala dengan masing-masing 2 hingga 5 item untuk menilai hal yang sama [3]. Tes DASS meminta subjek untuk menilai seberapa banyak mereka telah menghadapi masing-masing kondisi yang disebutkan dalam seminggu terakhir menggunakan salah satu dari empat opsi yang disajikan dalam bentuk skala likert. Selain itu, untuk mendapatkan gambaran umum tentang tingkat stres maka skor dari masing-masing subskala ini dijumlahkan. Pada penelitian ini, parameter yang digunakan dalam pembuatan sistem yaitu parameter untuk pengukuran tingkat stres yang terdiri dari 14 pertanyaan.

2.2 Skala Likert

Skala likert digunakan untuk menilai sikap, pendapat, serta persepsi seseorang atau sekelompok orang tentang fenomena sosial [7]. Dengan skala likert, maka variabel yang akan dinilai diuraikan sebagai indikator variabel. Lalu indikator tadi dijadikan sebagai titik tolak dalam menyusun item-item instrumen yang bisa berupa pernyataan atau pertanyaan. Nilai setiap item instrumen yang memakai skala likert mempunyai susunan berasal sangat positif sampai sangat negatif, berupa pilihan ganda ataupun *checklist*.

2.3 UML

Salah satu alat yang sangat digunakan dalam bidang perancangan sistem berorientasi objek adalah *Unified Modelling Language* atau sering disingkat dengan UML [13]. Banyak diagram pada UML dapat mengakomodasi perspektif yang berbeda pada perangkat lunak yang sedang dirancang atau dikembangkan. Dengan menggunakan diagram ini, seseorang dapat menyampaikan ide, memunculkan konsep dan kemungkinan baru, menguji konsep dan membuat prediksi, serta memahami struktur dan relasi-relasinya [8].

2.4 Database

Database adalah seperangkat data terorganisir yang disimpan di komputer dan dapat digunakan untuk menghasilkan informasi dengan diproses atau diubah oleh perangkat lunak. Tipe data, struktur data, dan batasan data semuanya termasuk dalam definisi *database* [9].

2.5 Framework Codeigneter

Framework Codeigneter adalah kerangka kerja MVC (*Model, View, Controller*) berbasis PHP yang tersedia sebagai perangkat lunak sumber terbuka dan digunakan untuk membuat situs web dinamis. *Codeigneter* dikembangkan sebagai alat untuk memfasilitasi dan mempercepat pekerjaan pengembang web sekaligus meningkatkan sistem keamanannya [10].

2.6 Metode Waterfall

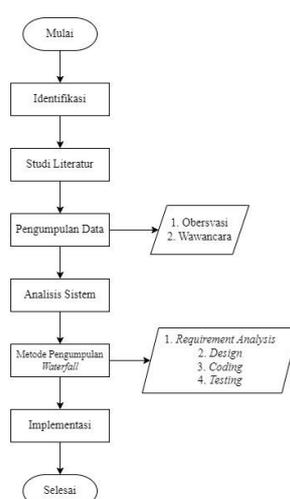
Metode *waterfall* adalah salah satu metode pengembangan sistem yang setiap tahap harus menunggu tahap sebelumnya untuk bisa melanjutkan ke tahap berikutnya secara berurutan. Model pengembangan sistem ini mengikuti jalur linier dari tahap perencanaan yang dilakukan pertama dan melalui tahap pemeliharaan sebagai tahap terakhir. Sebelum tahap yang terakhir selesai dilakukan, tahap selanjutnya tidak dapat dimulai dan tidak dapat mengulangi tahapan sebelumnya [11]

2.7 Blackbox Testing

Dalam pengujian *blackbox testing*, persyaratan fungsional perangkat lunak adalah penekanan utama. Penguji dapat memberikan serangkaian keadaan masukan dan menjalankan pengujian pada persyaratan fungsional program [12].

3 Metodologi Penelitian

3.1 Alur Penelitian



Gambar. 1. Alur Penelitian

Dari Gambar 1 dapat dijelaskan tahapan penelitian yang dilakukan sebagai berikut.

1. Identifikasi Masalah

Dalam penelitian ini, penulis terlebih dahulu melakukan identifikasi dari suatu permasalahan, dimana masalah yang teridentifikasi yaitu peningkatan stres pada mahasiswa Fakultas Ilmu Komputer (FIK) UPN “Veteran” Jakarta selama masa pandemi.

2. Studi Literatur

Dalam penyusunan tugas akhir ini, penulis terlebih dahulu melakukan studi literatur, dimulai dengan menggunakan data dan informasi yang berkaitan dengan pembangunan sistem, pemahaman tentang sistem, hingga memahami tentang DASS 42 yang akan menjadi alat ukur tingkat stres pada mahasiswa serta pemahaman dalam proses perancangan sistem yang akan dibangun. Sumber literatur yang digunakan dalam penulisan tugas akhir ini dapat dilihat pada daftar pustaka.

3. Pengumpulan Data

Dalam penyusunan tugas akhir ini, penulis mengumpulkan data dan detail yang diperlukan untuk membuat sistem pengukuran tingkat stres pada data mahasiswa FIK UPN “Veteran” Jakarta.

4. Analisis sistem

Penulis melakukan studi terhadap sistem yang ada dengan memeriksa data yang dikumpulkan selama pengumpulan data mahasiswa, hingga ditemukan hasil tingkat stres pada masing-masing data mahasiswa berdasarkan hasil dari setiap tes yang dilakukan.

5. Metode Pengembangan *Waterfall*

Pada membangun sistem *StressLevel*, penulis menggunakan metode pengembangan *Waterfall* yang memiliki beberapa tahap, yaitu:

a. Requirement

Pada tahap ini, penulis mengevaluasi semua kebutuhan sistem, termasuk fungsional dan non-fungsional. Persyaratan fungsional adalah spesifikasi yang mencakup operasi untuk mampu dilakukan oleh sistem seperti menginput jawaban tes yang dibutuhkan untuk nantinya bisa menampilkan hasil tingkat stres dari tes yang dilakukan. Kebutuhan non-fungsional adalah kebutuhan yang lebih mementingkan karakteristik perilaku sistem seperti tampilan dari sistem yang akan dibuat.

b. Design

Dalam memvisualisasikan dan menetapkan proses yang harus diikuti serta data yang dibutuhkan untuk sistem *StressLevel* yang akan dibangun, penulis memasukkan desain diagram UML ke dalam desain sistem serta membuat rancangan *user interface* dari sistem. Tugas akhir ini menggunakan desain UML untuk membuat *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram* untuk bisa memudahkan penulis dalam membangun sistem *StressLevel* ini.

c. Coding

Dengan mengubah sistem dari langkah sebelumnya menjadi program kode yang dapat dibaca oleh mesin, penulis mempraktikkan desain dari UML tersebut. Program komputer kemudian akan digabungkan dengan kode program untuk menghasilkan sistem *StressLevel* sesuai dengan yang sudah dirancang pada tahap sebelumnya. *Visual Studio Code* dan pemrograman PHP digunakan untuk merancang pengkodean dari sistem *StressLevel* berbasis web tersebut.

d. Testing

Setelah tahap *coding* selesai, penulis melakukan pengujian sistem. Metode *blackbox testing* digunakan dalam tahap *testing* ini untuk mengetahui apakah sistem *StressLevel* yang dibangun dapat menjalankan fungsinya, mengetahui apakah terdapat kekurangan dalam proses perancangan kode serta sistem yang dihasilkan apakah masih memiliki kekurangan atau tidak. Pengguna yang merupakan mahasiswa dipilih secara acak untuk menguji sistem *StressLevel* ini.

6. Implementasi

Implementasi adalah menerapkan apayang sudah dibuat sebelumnya[14]. Sistem *StressLevel* dapat diimplementasikan di instansi terkait yaitu FIK UPN “Veteran” Jakarta setelah dilakukan uji dan diperoleh hasil bahwa sistem yang dibangun sudah berjalan sesuai dengan yang diharapkan.

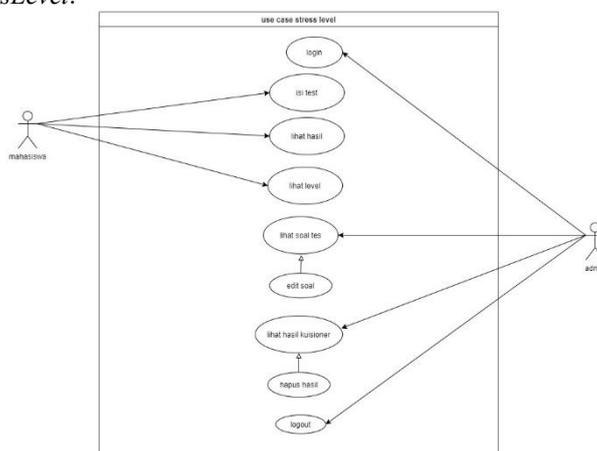
4 Tinjauan Pustaka

4.1 Analisis Sistem Usulan

Pengukuran tingkat stres pada mahasiswa FIK UPN “Veteran” Jakarta masih dilakukan secara manual. Dimana pada hasil tes yang dilakukan dengan menggunakan *Google Form* dan perhitungan secara manual yaitu menggunakan rumus pada file *spreadsheet* responden. Karena proses perhitungan ini masih dilakukan dengan kurang efektif, mahasiswa tidak bisa mengetahui langsung tingkat stres yang sedang mereka alami. Oleh sebab itu, sistem ini dibuat untuk mengubah sistem yang dilakukan secara manual tersebut menjadi terkomputasi sehingga hasil yang diharapkan yaitu mahasiswa dapat mengetahui tingkat stres pada pribadi masing-masing secara lebih efisien.

4.1.1 Use Case Diagram

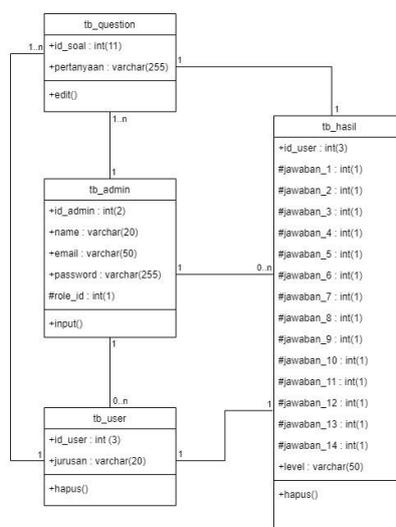
Dalam menggambarkan interaksi antara aktor dan kegunaannya pada sistem *StressLevel*, maka terlebih dahulu dibuat *use case diagram* dari sistem ini. Pada gambar 2 dapat dilihat gambaran fungsi dari aktor mahasiswa dan admin terhadap sistem *StressLevel*.



Gambar. 2. Use case diagram *StressLevel*

4.1.2 Class Diagram

Dalam menggambarkan struktur sistem serta penjelasan dan hubungan dari kelas yang terdapat dalam sistem *StressLevel* maka dibuatlah rancangan *class diagram* seperti pada gambar 3 dibawah ini.



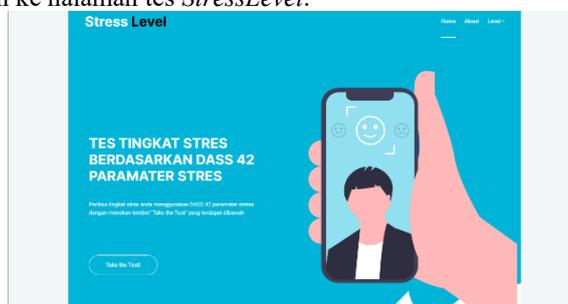
Gambar. 3. Class diagram *StressLevel*

4.1.3 Implementasi Sistem

Setelah tahap perancangan dari pembuatan sistem selesai dilakukan, maka penulis melanjutkan ke dalam tahap implementasi sistem *StressLevel*. Adapun tampilan dari sistem yang dibuat adalah sebagai berikut.

a. Halaman *Home StressLevel*

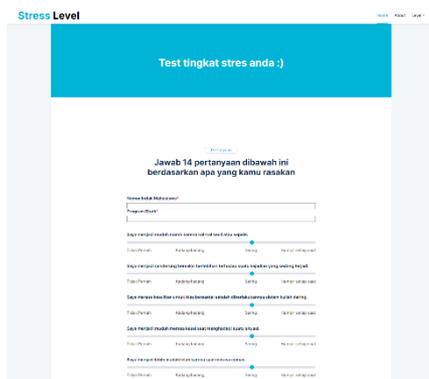
Pada halaman ini, *user* yakni mahasiswa akan melihat tampilan *home* dari sistem *StressLevel* setelah berhasil mengakses URL dari sistem tersebut. Pada halaman ini mahasiswa bisa memilih tombol ‘*Take the Test!*’ untuk berpindah ke halaman tes *StressLevel*.



Gambar. 4. Halaman *Home StressLevel*

b. Halaman Tes *StressLevel*

Pada halaman ini, mahasiswa akan mengisi jawaban dari setiap pertanyaan yang dibutuhkan. Untuk jawaban dari setiap soal tes harus diisi sesuai dengan keadaan yang sedang mereka rasakan sejujur-jujurnya untuk bisa menghasilkan tes yang sesuai dengan yang dialami.



Gambar. 5. Halaman Tes *StressLevel*

c. **Halaman Hasil *StressLevel***

Pada halaman ini, sistem akan menampilkan hasil tingkat stres yang dialami oleh mahasiswa berdasarkan setiap nilai jawaban yang telah diisi mahasiswa pada halaman tes sebelumnya.



Gambar. 6. Halaman Hasil *StressLevel*

d. **Halaman Level *StressLevel***

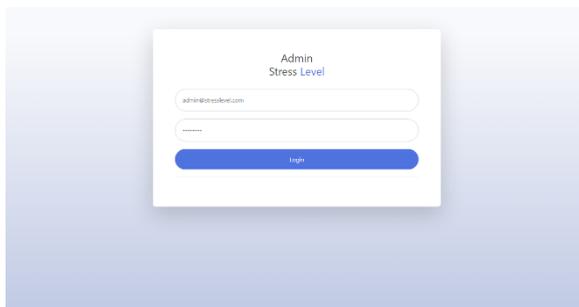
Setelah mengetahui hasil tingkat stres yang dialami, mahasiswa juga dapat melihat beberapa saran yang dapat dilakukan untuk mengendalikan tingkat stres sesuai dengan tingkatannya pada halaman level.



Gambar. 7. Halaman *StressLevel* Berdasarkan Tingkat Stres

e. **Halaman *Login Admin StressLevel***

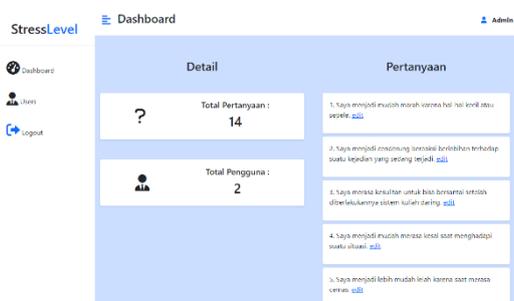
Pada halaman ini, admin akan melihat tampilan halaman *login* admin setelah berhasil mengakses halaman khusus admin. Untuk bisa mengakses ke halaman *Dashboard*, admin harus memasukkan email dan *password* yang benar.



Gambar. 8. Halaman Login Admin StressLevel

f. Halaman Dashboard Admin StressLevel

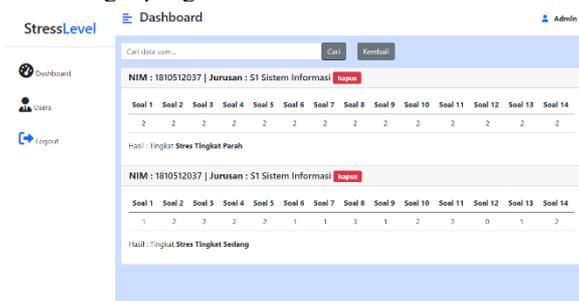
Pada halaman ini, admin dapat melihat daftar pertanyaan yang diperlukan pada sistem StressLevel.



Gambar. 9. Halaman Dashboard Admin StressLevel

a. Halaman Dashboard Admin StressLevel

Pada halaman ini, admin dapat melihat data seluruh mahasiswa yang telah masuk ke dalam sistem StressLevel. Selain itu, admin juga dapat menghapus data mahasiswa tersebut dari sistem. Hak ini hanya bisa dilakukan oleh admin sebagai pengelola sistem StressLevel.



Gambar. 10. Halaman Users Admin StressLevel

4.2 Blackbox Testing

Pada tahap terakhir setelah sistem dibuat akan dilakukan pengujian sistem dari sisi user atau mahasiswa dengan menjalankan setiap fitur dari sistem untuk melihat apakah berjalan dengan sesuai yang seharusnya. Sistem harus merespons dengan umpan balik yang tepat dan relevan. Adapun hasil dari blackbox testing ini tertera pada Tabel 1 dan Tabel 2 berikut.

Tabel 1. Blackbox Testing Halaman Mahasiswa

Modul	Modul yang Diuji	Keluaran yang Diharapkan	Hasil yang Didapatkan	Status	
Halaman <i>Home</i>	Halaman diakses	URL	Halaman awal ditampilkan	Halaman awal ditampilkan	Berhasil
	Menekan <i>About</i>	tombol	Halaman <i>about</i> ditampilkan	Halaman <i>about</i> ditampilkan	Berhasil
	Menekan <i>Take the Test!</i>	tombol	Halaman tes ditampilkan	Halaman tes ditampilkan	Berhasil
Halaman Tes Tingkat Stres	Menekan <i>Submit</i>	tombol	Halaman hasil tes ditampilkan	Halaman hasil tes ditampilkan	Berhasil
Halaman Level Tingkat Stres	Menekan <i>Level, Normal</i>	tombol memilih	Halaman tingkat stres level Normal ditampilkan	Halaman tingkat stres level Normal ditampilkan	Berhasil
	Menekan <i>Level, Ringan</i>	tombol memilih	Halaman tingkat stres level Ringan ditampilkan	Halaman tingkat stres level Ringan ditampilkan	Berhasil
	Menekan <i>Level, Sedang</i>	tombol memilih	Halaman tingkat stres level Sedang ditampilkan	Halaman tingkat stres level Sedang ditampilkan	Berhasil
	Menekan <i>Level, Parah</i>	tombol memilih	Halaman tingkat stres level Parah ditampilkan	Halaman tingkat stres level Parah ditampilkan	Berhasil
	Menekan <i>Level, Sangat Parah</i>	tombol memilih	Halaman tingkat stres level Sangat Parah ditampilkan	Halaman tingkat stres level Sangat Parah ditampilkan	Berhasil

Tabel 2. Blackbox Testing Halaman Admin

Modul	Modul yang Diuji	Keluaran yang Diharapkan	Hasil yang Didapatkan	Status	
Halaman <i>Login</i>	Halaman diakses	URL	Halaman <i>login</i> admin ditampilkan	Halaman <i>login</i> admin ditampilkan	Berhasil
	Menekan <i>Login</i>	tombol	Halaman utama admin ditampilkan	Halaman utama admin ditampilkan	Berhasil
Halaman <i>Dashboard Admin</i>	Menekan edit soal	tombol	Halaman edit soal ditampilkan	Halaman edit soal ditampilkan	Berhasil
	Menekan <i>Users</i>	menu	Halaman <i>users</i> ditampilkan	Halaman <i>users</i> ditampilkan	Berhasil
	Menekan <i>logout</i>	menu	Halaman <i>login</i> ditampilkan	Halaman <i>login</i> ditampilkan	Berhasil
Halaman <i>Users Admin</i>	Menekan hapus satu data	tombol	Memunculkan <i>alert</i> "Apakah anda yakin ingin menghapus result user?"	Memunculkan <i>alert</i> "Apakah anda yakin ingin menghapus result user?"	Berhasil
	Menekan <i>alert</i>	tombol	Jika iya, menampilkan halaman <i>users</i> dengan data yang sudah berhasil terhapus. Jika tidak, menampilkan halaman <i>users</i> tanpa perubahan apapun.	Jika iya, menampilkan halaman <i>users</i> dengan data yang sudah berhasil terhapus. Jika tidak, menampilkan halaman <i>users</i> tanpa perubahan apapun.	Berhasil

5 Kesimpulan

Kesimpulan ini dapat dibuat berdasarkan penelitian yang telah dilakukan, yaitu:

1. Alat ukur DASS 42 parameter stres berfungsi untuk mendeteksi atau mengukur tingkat stres dan fitur level untuk mengidentifikasi cara mengatasi stres yang sedang dialami serta dapat digunakan untuk membangun sistem *StressLevel* secara efektif.

2. Sistem *StressLevel* yang telah dibuat memanfaatkan *framework codeigneter* serta PHP sehingga memudahkan penulis dalam pembuatan sistem.
3. Berdasarkan *blackbox testing* yang telah dilakukan terhadap sistem yang dibuat yaitu sistem *StressLevel* menunjukkan bahwa seluruh fitur yang ada pada sistem sudah mendapatkan status berhasil, sehingga sistem *StressLevel* sudah berfungsi sesuai dengan yang diharapkan hanya saja masih terdapat kekurangan dalam implementasi tampilan web.
4. Hasil dari penelitian yang telah dilakukan ini adalah terbangunnya sistem *StressLevel* yang dapat membantu mahasiswa FIK UPN “Veteran” Jakarta dalam mengukur tingkat stres yang dialami dengan lebih efektif yaitu meningkatkan kepedulian terhadap kesehatan mental terutama pada kesehatan stres pribadi mahasiswa.
5. Berdasarkan 273 data mahasiswa yang telah masuk ke dalam sistem diketahui bahwa tingkat stres yang dialami oleh mahasiswa pada masa pandemi terbanyak berada pada tingkat stres parah. Banyaknya mahasiswa dari masing-masing tingkat stres yaitu sebanyak 54 data mahasiswa berada pada tingkat stres normal, 54 data mahasiswa pada tingkat stres ringan, 46 data mahasiswa pada stres tingkat sedang, 68 data mahasiswa pada stres tingkat parah, dan 51 data mahasiswa pada stres tingkat sangat parah.

Referensi

- [1] Deliviana, E., Erni, M. H., Hilery, P. M., & Naomi, N. M. (2020). PENGELOLAAN KESEHATAN MENTAL MAHASISWA BAGI OPTIMALISASI PEMBELAJARAN ONLINE DI MASA PANDEMI COVID-19. *Jurnal Selaras : Kajian Bimbingan Dan Konseling Serta Psikologi Pendidikan*, 3(2).
- [2] *Infografik Masalah Psikologis Terkait Pandemi COVID-19 di Indonesia*. (2020). Perhimpunan Dokter Spesialis Kedokteran Jiwa Indonesia. <http://www.pdskji.org/home>
- [3] Damanik, E. D. (2006). Pengujian Reliabilitas, Validitas, Analisis item dan Pembuatan Norma Depression Anxiety Stress Scale (DASS). *Universitas Indonesia*.
- [4] Syafitri, E. N. (2019). ANALISIS KUALITATIF WARAS (RESPONSIVE WEB TO ACCESS FAST RESPONSE FOR ATTENDING STRESS, ANXIETY AND DEPRESSION AMONG STUDENTS) SEBAGAI SISTEM PELAYANAN KESEHATAN MENTAL MAHASISWA UNRIYO. *Seminar Nasional UNRIYO*.
- [5] Simanullang, R., & Situmorang, P. C. (2020). *Managemen Stres di Tengah Dampak Covid-19*. Guepedia.
- [6] Mahmud, R., & Ayun, Z. (2021). Stress, Koping dan Adaptasi Teori dan Pohon Masalah Keperawatan. *Jurnal Indigenous*, 1(2).
- [7] Sugiyono. (2017). *Metode Penelitian Kuantitatif, Kualitatif Dan R&D*.
- [8] Henderi. (2009). Object Oriented Modelling with Unified Modelling Language (Uml). 2016, June.
- [9] Yudhanto, Y., & Prasetyo, H. A. (2019). *Mudah Menguasai Framework Laravel*. Jakarta : PT. Gramedia, 2019.
- [10] Sulistyono, G. B., & Widodo, P. (2020). *Membuat Web dengan Framework Codeigneter (Studi Kasus Sistem Informasi Perpustakaan) (Pertama)*. Yogyakarta : Graha Ilmu, 2020.
- [11] Aceng Abdul Wahid. (2020). Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi. *Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK*, November.
- [12] Khan, Mohd. E. (2011). Different Approach to Blackbox Testing Technique for Finding Error, *International Journal of Software Engineering & Applications (IJSEA)*. *International Journal of Software Engineering and Its Applications*, 2(4).
- [13] Muliawati, A., & Purabaya, R. H. (2021, July). Perancangan sistem informasi tracking surat kependudukan pada kantor desa sukamanah berbasis website. In *Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya (Vol. 2, No. 1, pp. 20-30)*.
- [14] Mardiyah, H. A., Krisnanik, E., & Irmanda, H. N. (2021, July). Sistem Informasi Customer Relationship Management pada Budidaya Jamur Tiram di Tangerang Selatan Berbasis Web. In *Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya (Vol. 2, No. 1, pp. 51-60)*.