

## ANALISIS KUALITAS SISTEM LAYANAN PENILAIAN BUKU PENDIDIKAN AGAMA (SiLPBPA) MENGGUNAKAN METODE *SYSTEM USABILITY SCALE* (SUS) DAN PENGEMBANGANNYA MENGGUNAKAN *SCRUM* BERBASIS WEB

Asep Erlan Maulana<sup>1</sup>, Abu Khalid Rivai<sup>2</sup>, Sarwani Sarwani<sup>3</sup>

<sup>1,2</sup>Pascasarjana Magister Komputer Teknik Informatika  
Sekolah Tinggi Manajemen Informatika dan Komputer Eresha

<sup>3</sup>Pascasarjana Magister Manajemen

Universitas Pamulang Tangerang Selatan, Banten, Indonesia

<sup>1</sup>ceperlan@gmail.com, <sup>2</sup>dosen40042@eresha.ac.id, <sup>3</sup>sarwani3082@yahoo.com

### ABSTRACT

SiLPBPA version 1.0 is implemented in the Religious Education Book Assessment Service activities. However, there are many obstacles and problems, namely that some system functions do not match the needs that support their activities. The purpose of this research is to conduct quality analysis and develop SiLPBPA. The methodology for conducting quality analysis is the System Usability Scale (SUS), and its development uses the Agile method with a Scrum framework. The results show that the system's quality gets a score of 71, which means the system is Acceptable, is in Predicate C, and with a good rating. However, some users gave some suggestions on the questionnaire. These suggestions are evaluated, which are used as the basis for system development. The results of developing using Scrum can be completed within 65 hours of the target time of 70 hours shown on the Burndown Chart. Some system functions can be completed properly. This was proven at the Sprint Review event, the Scrum Team and Stakeholders who attended approved the Product Increments that had been developed in all Sprints. So that SiLPBPA can be used in the next Education Book Assessment service activities.

**Keywords:** *SiLPBPA, System Usability Scale (SUS), Scrum*

### ABSTRAK

SiLPBPA versi 1.0 diimplementasikan pada kegiatan Layanan Penilaian Buku Pendidikan Agama. Namun, terdapat banyak kendala dan permasalahan yaitu beberapa fungsi pada sistem tidak sesuai dengan kebutuhan yang mendukung kegiatannya. Tujuan penelitian ini adalah untuk melakukan analisis kualitas dan melakukan pengembangan sistem. Metodologi untuk melakukan analisis kualitas adalah *System Usability Scale* (SUS) dan pengembangannya menggunakan metode *Agile* dengan kerangka kerja *Scrum*. Hasil penelitian menunjukkan pada kualitas sistem mendapatkan skor 71 yang berarti sistem dapat diterima (*Acceptable*), ada pada Predikat C, dan dengan *Rating Good*. Namun beberapa *user* memberikan beberapa saran pada kuesioner. Saran tersebut dilakukan evaluasi yang dijadikan dasar untuk melakukan pengembangan sistem. Hasil dari pengembangan menggunakan *Scrum* bisa diselesaikan dalam waktu 65 jam dari target waktu 70 jam yang terlihat pada *Burndown Chart*. Beberapa fungsi sistem dapat selesai dengan baik. Hal ini terbukti pada acara *Sprint Review*, Tim *Scrum* dan *Stakeholder* yang hadir menyetujui *Product Increment* yang sudah dikembangkan pada semua *Sprint*, sehingga SiLPBPA dapat digunakan pada kegiatan layanan Penilaian Buku Pendidikan selanjutnya.

**Kata Kunci:** *SiLPBPA, System Usability Scale (SUS), Scrum*

## I. PENDAHULUAN

Dalam Peraturan Menteri Agama Republik Indonesia Nomor 9 Tahun 2018 [1] tentang Buku Pendidikan Agama disebutkan bahwa Badan Litbang dan Diklat Kementerian Agama melaksanakan penilaian buku pendidikan agama. Dalam melaksanakan tugasnya Badan Litbang dan Diklat Kementerian Agama, mengeluarkan SK Kepala Badan Litbang dan Diklat Kementerian Agama Nomor 51 Tahun 2018 [2] tentang penulisan, penilaian, dan penerbitan buku Pendidikan Agama. Di dalam SK tersebut menunjuk Puslitbang Lektur, Khazanah dan Manajemen Organisasi sebagai *stakeholder* yang melaksanakan penilaian buku Pendidikan Agama.

Pada awalnya, kegiatan ini dilakukan secara manual. *Stakeholder* bertemu dalam satu waktu dan satu tempat. Para penerbit datang ke kantor Puslitbang Lektur, Khazanah dan Manajemen Organisasi untuk melakukan pendaftaran dengan menyerahkan kelegalan perusahaan dan buku dalam bentuk *hardcopy*, kemudian dilakukan penilaian oleh para penilai sampai buku layak terbit dan mendapatkan lembar persetujuan terbit. Hal ini sangat merepotkan *stakeholder* yang terlibat dalam kegiatan ini. Oleh karena itu, tahun 2018 Puslitbang Lektur, Khazanah Keagamaan dan Manajemen Organisasi melakukan pengembangan sebuah aplikasi. Aplikasi ini dinamakan SiLPBPA (Sistem Layanan Penilaian Buku Pendidikan Agama). Versi 1.0 dari sistem ini hanya untuk melayani pendaftaran dan penerbit menyerahkan kelegalan perusahaan secara daring. Pada implementasinya, sistem ini banyak kendala dan permasalahan didalam penggunaannya. Beberapa fungsi sistem tidak sesuai dengan kebutuhan yang mendukung pada kegiatannya.

Dari permasalahan tersebut maka perlu dilakukan analisis kualitas layanan dan pengembangan sistem. Dalam melakukan analisis kualitas layanan dari SiLPBPA diperlukan tes kegunaan untuk melakukan evaluasi dari sebuah produk perangkat lunak yang sudah dibuat [3]. Di dalam ilmu UX (*User Experience*) terdapat pembahasan pengukuran tingkat kegunaan (*Usability*). Artinya kegiatan ini dilakukan untuk mengukur tingkat kegunaan sebuah sistem yang dilihat dari berbagai aspek kepuasan pengguna [4]. Salah satu dalam melakukan pengujian kegunaan adalah menggunakan *System Usability Scale* (SUS).

*System Usability Scale* (SUS) adalah satu metode untuk melakukan evaluasi sebuah produk perangkat lunak dengan pertimbangan jumlah sampel yang kecil, waktu dan biaya yang memberikan hasil yang memadai [5]. Menurut Martoyo dan Falahah [6], *System Usability Scale* (SUS) merupakan sebuah teknik pengujian kegunaan sebuah perangkat lunak yang dalam proses pengerjaannya melibatkan pengguna akhir. Bisa membedakan kegunaan perangkat lunak dengan biaya murah dan terjangkau serta dapat diandalkan hasilnya meskipun dengan jumlah sampelnya yang lebih sedikit [7].

Setelah melakukan pengujian di dalam penggunaan sistem, tahap selanjutnya adalah melakukan pengembangan sistem dengan kerangka kerja *Scrum*. *Scrum* dipilih karena salah satu kerangka kerja yang populer dari metode pengembangan sistem *Agile* [8]. *Scrum* dapat mengatasi masalah kompleks, menghantarkan produk dengan setinggi mungkin secara produktif dan kreatif [9]. Di dalam penelitian Murdiani, Yudhana dan Sunardi [10], yang dirilis oleh *The Standish Group* antara tahun 2002-2010 menunjukkan metode *Agile* lebih sukses tingkat keberhasilannya sebanyak 42 % dibandingkan dengan metode lain seperti *Waterfall* 14 %. Kerangka kerja *Scrum* banyak digunakan di dalam pengembangan perangkat lunak [11][12][13]. Dapat lebih cepat memprediksi untuk mengetahui resiko dan kualitas dari pengembangan perangkat lunak [14]. Masalah yang terjadi pada proses pengembangan bisa diketahui secara cepat [15]. Menyelesaikan pengembangan sistem secara *realtime* dengan cepat. Umpan balik dari *user* dengan cepat dan dapat merilis produk di setiap akhir *sprint*-nya [16]. Bersifat ringan, sederhana dan dengan pendekatan *bottom up* [17].

Beberapa penelitian yang melakukan kajian tentang menguji kualitas, hanya sampai pengujian *usability* perangkat lunaknya saja tidak sampai tahap pengembangan. Oleh karena itu, penulis melakukan penelitian ini bertujuan untuk analisis kualitas *usability* aplikasi SiLPBPA versi 1.0 menggunakan *System Usability Scale* (SUS) dan melakukan pengembangan sistem dari hasil uji kualitas menggunakan *Scrum*. Agar sistem tersebut dapat segera digunakan langsung dengan baik dan efektif yang sesuai dengan kebutuhan

penggunanya pada kegiatan yang segera dilaksanakan.

## II. METODOLOGI

Penelitian ini diawali dengan melakukan observasi selama kegiatan layanan penilaian buku Pendidikan Agama sedang berlangsung dari awal bulan Januari sampai bulan Desember 2019. Kemudian melakukan studi pustaka dengan mengumpulkan dan mengkaji beberapa artikel dari jurnal ilmiah nasional maupun jurnal internasional. Beberapa buku, artikel hasil publikasi pada *conference*, dan sumber-sumber terpercaya dari internet yang dapat mendukung dalam proses penelitian ini.

Metode pengumpulan data yang digunakan adalah menggunakan kuesioner dari instrumen data kuesioner *System Usability Scale* (SUS). Kuesioner disebarikan secara daring menggunakan *Google Form* kepada pengguna yang terlibat didalam sistem yaitu *Admin*, *Operator* dan *Penerbit*. Kuesioner terdiri dari 10 item pertanyaan dengan menggunakan skala 5 jawaban.

Tabel 1. Instrumen data kuesioner SUS [18]

Kode	Instrumen Pertanyaan	Skala
Q1	1. Saya berpikir akan sering menggunakan sistem ini.	1-5
Q2	2. Saya rasa sistem ini rumit untuk digunakan.	1-5
Q3	3. Saya rasa sistem ini mudah digunakan.	1-5
Q4	4. Saya membutuhkan bantuan dari orang lain atau teknisi untuk dapat menggunakan sistem ini.	1-5
Q5	5. Saya rasa fitur dari berbagai fungsi sistem ini berjalan sebagaimana mestinya.	1-5
Q6	6. Saya menemukan sistem ini tidak konsisten.	1-5
Q7	7. Saya rasa kebanyakan orang akan mudah belajar untuk menggunakan sistem ini dengan sangat cepat.	1-5
Q8	8. Saya rasa sistem ini sangat rumit untuk digunakan.	1-5
Q9	9. Saya rasa tidak ada hambatan dalam menggunakan sistem ini.	1-5
Q10	10. Saya perlu membiasakan diri terlebih dahulu sebelum bisa memulai menggunakan sistem ini.	1-5

Tabel 2. Skor jawaban pertanyaan kuesioner SUS

Jawaban	Skor
Sangat Tidak Setuju (STS)	1
Tidak Setuju (TS)	2
Ragu-Ragu (RG)	3
Setuju (S)	4
Sangat Setuju	5

Kalimat dengan pertanyaan positif (pada instrumen pertanyaan ganjil. Kode: Q1, Q3, Q5, Q7, Q9), skor kontribusinya adalah posisi skala dikurangi 1. Kalimat dengan pertanyaan negatif (pada instrumen pertanyaan genap. Kode: Q2, Q4, Q6, Q8, Q10), skor kontribusinya adalah 5 dikurangi posisi skala [19].

$$Skor\ SUS = ((Q1 - 1) + (5 - Q2) + (Q3 - 1) + (5 - Q4) + (Q5 - 1) + (5 - Q6) + (Q7 - 1) + (5 - Q8) + (Q9 - 1) + (5 - Q10)) * 2,5$$

Nilai keseluruhan *Usability* didapat dari skor kontribusi yang berkisar antara 0 sampai 4 dikalikan 2,5. Persamaan 1 adalah rumus perhitungan skor SUS secara keseluruhan [20]:

$$\bar{X} = \frac{\sum \chi}{n} \quad (1)$$

$$\begin{aligned} \bar{X} &= \text{skor rata - rata} \\ \sum \chi &= \text{jumlah skor SUS} \\ n &= \text{jumlah responden} \end{aligned}$$

Nilai keseluruhan Skor diperoleh dari rata-rata skor SUS individual [19]. Rentang skor SUS yang dimiliki adalah dari 0 sampai 100 [21]. Ada 2 cara dalam melakukan penentuan hasil akhir penilaian [22]. (1) Penentuan hasil yang menggunakan skor *Percentile Rank* [23]. Menurut Ependi [24] skor *Percentile Rank* ditentukan ke dalam *grade* penilaian yang ditunjukkan pada Tabel 3.

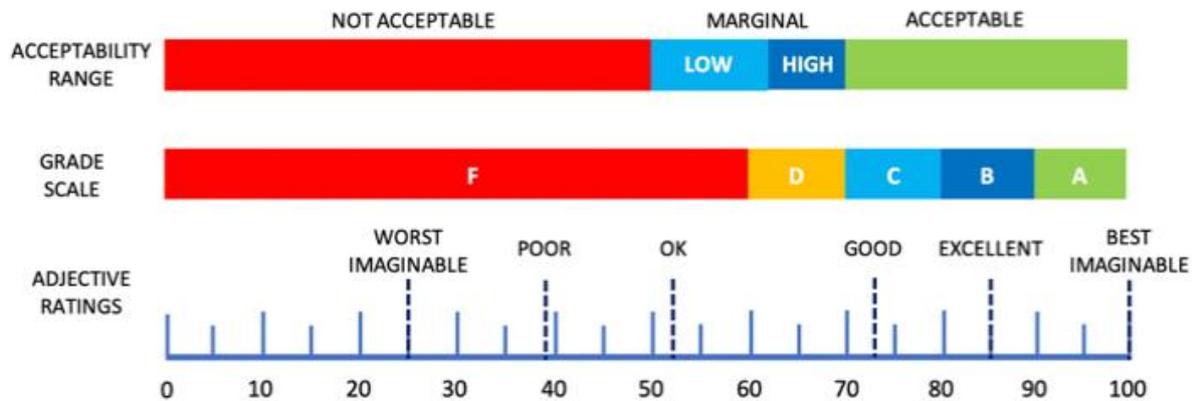
Tabel 3. Grade penilaian skor SUS percentile rank

Grade	Skor
A	Skor lebih besar atau sama dengan 80,3
B	Skor lebih besar sama dengan 74 dan lebih kecil 80,3
C	Skor lebih besar 68 dan lebih kecil 74
D	Skor lebih besar sama dengan 51 dan lebih kecil 68
E	Skor lebih kecil dari 51

(2) Penentuan hasil yang menggunakan *Acceptability Range*, *Grade Scale*, dan

*Adjective Rating* [25]. Terdapat 3 kategori penerimaan anggota pada *Acceptability Range* yaitu *acceptable*, *marginal*, *not acceptable*. Pada *Grade Scale* ada 6 skala tingkat penerimaan pengguna yaitu A,B,C,D, dan F.

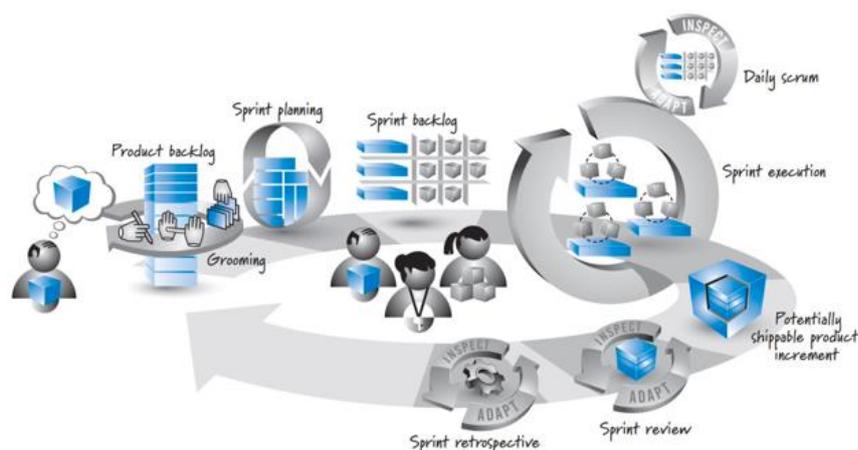
dan tingkat penerimaan pengguna pada *Adjective Rating* adalah *worst imaginable*, *poor*, *good*, *excellent*, dan *best imaginable*.



Gambar 1. Kategori skor SUS *acceptability range*, *grade scale*, dan *adjective rating* [25]

Pengembangan sistem menggunakan metode *Agile* dengan kerangka kerja *Scrum*. Menurut Rubin [26] ada beberapa tahapan dalam pelaksanaan *Scrum* antara lain (Gambar 2): (1) *Product Backlog Grooming* adalah kegiatan untuk membuat, mengidentifikasi, mengatur prioritas dan memperbaiki *Product Backlog Item*. (2) *Sprint Planning* adalah kegiatan kolaboratif (oleh seluruh anggota tim *Scrum*) untuk memproses perencanaan pekerjaan yang akan dilakukan dalam satu *sprint*. Output kegiatan ini adalah *Sprint Goal* dan *Sprint Backlog* (3) *Sprint Execution* kegiatan untuk memenuhi tujuan *Sprint* (*Sprint Goal*) yang dilakukan oleh Tim *Scrum*. Setiap *Sprint* waktu maksimum penyelesaiannya 1 bulan dengan durasi 8 jam. (4) *Daily Scrum*

adalah kegiatan yang dilakukan oleh *Development Team* untuk evaluasi, kendala aktifitas yang sudah dilakukan dan rencana ke depannya untuk mencapai *Sprint Goal* dengan durasi waktu 15 menit selama *sprint* berlangsung. (5) *Sprint Review* adalah kegiatan diskusi antara Tim *Scrum* dengan *stakeholder* (Pemegang Kepentingan) untuk melakukan *review* pekerjaan yang sudah diselesaikan oleh *Development Team* selama satu *sprint*. (6) *Sprint Retrospective* adalah kegiatan untuk menginspeksi dirinya sendiri (*Development Team*) dan membuat perencanaan agar pada *sprint* selanjutnya meningkatkan pekerjaan yang dilakukan. Durasi waktu penyelenggaraan dalam 1 bulan adalah 3 jam.



Gambar 2. Tahapan pelaksanaan *scrum* [26].

Alur tahapan keseluruhan dari penelitian ini adalah sebagai berikut:



Gambar 3. Alur proses penelitian

### III. HASIL DAN PEMBAHASAN

#### 3.1 Data Penelitian

Hasil pengumpulan data dari kuesioner pada tanggal 17 Maret 2019 adalah sebanyak 14

Responden yang terdiri dari *user Admin* dan *Operator* berjumlah 5 orang dan *user Penerbit* berjumlah 9 orang. Berikut adalah rincian data tersebut:

Tabel 4. Data kuisisioner dari responden

No	Responden	Skor Kuisisioner dari responden									
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	Responden 1	4	2	4	2	2	4	2	2	2	4
2	Responden 2	4	2	4	2	3	3	3	2	3	4
3	Responden 3	4	2	4	4	3	3	4	3	4	4
4	Responden 4	5	2	4	3	4	2	4	2	4	5
5	Responden 5	4	2	4	2	4	3	4	2	4	4
6	Responden 6	5	2	4	3	3	3	3	3	3	4
7	Responden 7	5	2	4	2	4	2	4	2	4	4
8	Responden 8	5	2	4	2	4	2	4	2	4	4
9	Responden 9	5	2	4	2	4	2	4	2	4	2
10	Responden 10	4	2	4	2	4	3	3	2	4	4
11	Responden 11	4	2	4	2	2	2	4	2	4	4
12	Responden 12	4	2	4	2	4	2	4	2	4	4
13	Responden 13	4	2	4	2	3	2	4	2	4	4
14	Responden 14	4	2	4	2	4	2	4	2	4	2

### 3.2 Melakukan Perhitungan dengan Rumus SUS

Data hasil kuesioner yang diperoleh dari responden pada Tabel 4 dilakukan perhitungan dengan rumus SUS per responden [19] dan keseluruhan [20]. Diketahui rumus perhitungan Skor SUS per responden adalah:

$$\text{Skor SUS} = ((Q1 - 1) + (5 - Q2) + (Q3 - 1) + (5 - Q4) + (Q5 - 1) + (5 - Q6) + (Q7 - 1) + (5 - Q8) + (Q9 - 1) + (5 - Q10)) * 2.5$$

Kemudian dari hasil perhitungan dengan rumus per responden dihitung dengan rumus secara keseluruhan:

$$\bar{x} = \frac{50 + 60 + 58 + 68 + 68 + 58 + 73 + 73 + 78 + 65 + 65 + 70 + 68 + 75}{14}$$

$$\bar{x} = \frac{925}{14}$$

$$\bar{x} = 71$$

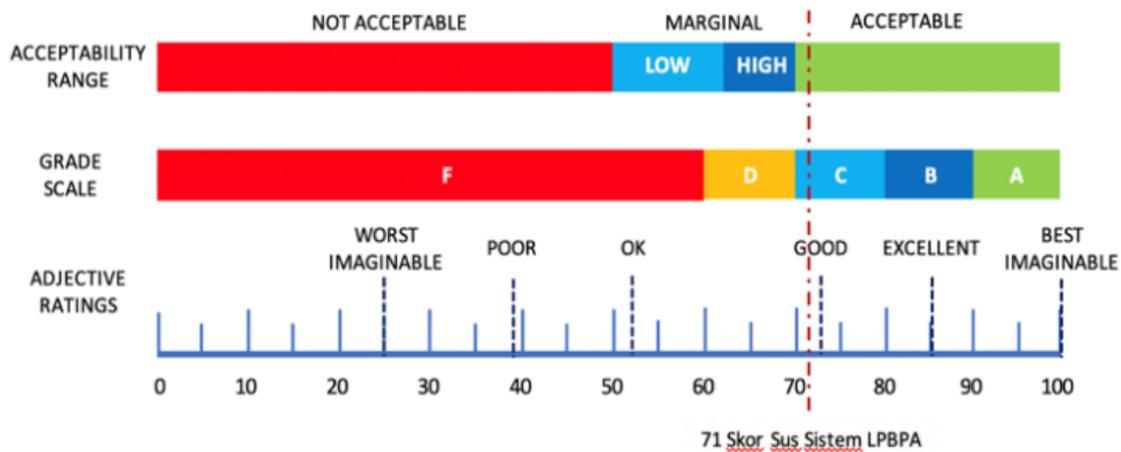
Tabel 5. Hasil perhitungan skor SUS

Responden	Skor Hasil Hitung SUS										Jumlah	Nilai (Jumlah x 2.5)
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
R1	3	3	3	3	1	1	1	3	1	1	20	50
R2	3	3	3	3	2	2	2	3	2	1	24	60
R3	3	3	3	1	2	2	3	2	3	1	23	58
R4	4	3	3	2	3	3	3	3	3	0	27	68
R5	3	3	3	3	3	2	3	3	3	1	27	68
R6	4	3	3	2	2	2	2	2	2	1	23	58
R7	4	3	3	3	3	3	3	3	3	1	29	73
R8	4	3	3	3	3	3	3	3	3	1	29	73
R9	4	3	3	3	3	3	3	3	3	3	31	78
R10	3	3	3	3	3	2	2	3	3	1	26	65
R11	3	3	3	3	1	3	3	3	3	1	26	65
R12	3	3	3	3	3	3	3	3	3	1	28	70
R13	3	3	3	3	2	2	3	3	3	1	26	65
R14	3	3	3	3	3	3	3	3	3	3	30	75
Skor rata-rata (Hasil Akhir)												71

### 3.3 Analisis Skor SUS

Hasil perhitungan dari skor *System Usability Scale* (SUS) menunjukkan Skor SUS akhir rata-rata sebesar 71. Berdasarkan penentuan hasil yang menggunakan skor *Percentile Rank* masuk ke dalam predikat C, sedangkan berdasarkan

penentuan hasil menggunakan *Acceptability Range* masuk ke dalam kategori *Acceptable*. Pada *Grade Scale* masuk ke dalam kategori C dan pada *Adjective Rating* masuk ke dalam kategori *Good*.



Gambar 4. Hasil nilai skor SUS pada *acceptability range*, *grade scale*, dan *adjective rating*

Dari hasil tersebut, maka dapat disimpulkan bahwa SiLPBPA dapat digunakan oleh pengguna. Skor SUS rata-rata 82 atau lebih berpotensi pengguna menjadi *Promoter* dan potensi pengguna menjadi *Deductor*, jika skor SUS rata-rata 67 atau kurang. Hal ini dilakukan karena skor SUS bisa dikorelasikan dengan *Net Promoter* menurut penjelasan Sauro [27]. Hasil skor SUS SiLPBPA adalah 71. Skor ini ada di antara *Promoter* dan *Deductor* namun cenderung penggunaanya *Deductor*, sehingga menyebabkan penurunan jumlah pengguna.

Penilaian Subyektif dari hasil nilai skor SUS pada sistem ini belum memenuhi kebutuhan pengguna. Hal ini berdasarkan 6 saran dari pengguna yang dicantumkan pada hasil kuesioner: (1) Di setiap tahapannya belum adanya petunjuk yang aplikatif (2) FAQ belum

ada (3) pada unggah berkas, tidak ada informasi jenis file dan *size* (4) Beberapa konten ditambahkan antara lain atribut lengkap pada komponen buku dengan melengkapi identitas dan id supaya tidak ada duplikat buku dengan judul yang sama (5) Seharusnya *user admin* dapat menghapus *user operator* (6) Seharusnya ditambahkan atribut Jenis Buku, Tahun Terbit, Jenjang Pendidikan dan kelas sesuai dengan metadata buku dari penerbit.

### 3.4 Melakukan Pengembangan Sistem dengan Scrum

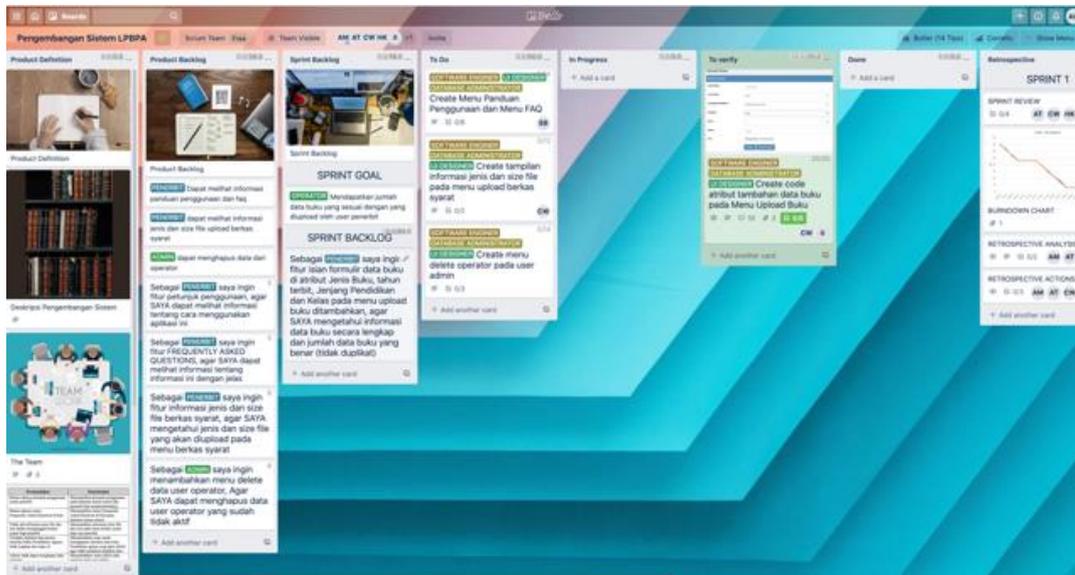
Berdasarkan hasil analisis skor SUS dan masukan dari pengguna maka penulis melakukan evaluasi pada sistem. Berikut adalah evaluasi permasalahan dan penyelesaiannya:

Tabel 6. Evaluasi sistem berdasarkan analisis skor SUS dan saran dari user

Permasalahan	Penyelesaian
Belum adanya petunjuk penggunaan untuk penerbit	Menampilkan petunjuk penggunaan pada halaman utama sistem dan penerbit bisa <i>download</i> -nya
Belum adanya menu <i>Frequently Asked Question</i> (FAQ)	Menampilkan menu <i>Frequently Asked Question</i> (FAQ) pada halaman utama sistem
Tidak ada informasi jenis file dan <i>size</i> dalam mengunggah berkas syarat bagi penerbit	Menampilkan informasi jenis file dan <i>size</i> pada menu berkas syarat bagi <i>user</i> Penerbit
Terdapat duplikat data karena identitas Buku Pendidikan Agama tidak lengkap dan tanpa id	Menambahkan isian untuk kelengkapan identitas data buku Pendidikan Agama yang akan dinilai agar tidak terjadinya duplikat data
<i>Admin</i> tidak dapat menghapus data operator	Menambahkan menu <i>delete</i> data operator pada <i>user Admin</i>
Pada data buku seharusnya ditambahkan atribut Jenis Buku, Tahun Terbit, Jenjang Pendidikan dan kelas sesuai dengan metadata buku dari penerbit	Menambahkan isian kolom data buku pada atribut Jenis Buku, Tahun Terbit, Jenjang Pendidikan dan kelas sesuai dengan <i>metadata</i> buku dari penerbit

Penyelesaian dari permasalahan pada tabel 6 adalah perlunya pengembangan pada SiLPBPA. Pengembangan dilakukan menggunakan kerangka kerja *Scrum* dan aplikasi *Trello*. *Trello* adalah alat kolaborasi visual yang digunakan sebagai papan tulis, *sticky notes* dan spidol *virtual* untuk menyelesaikan pengembangan SiLPBPA

bersama-sama secara fleksibel tanpa pembatasan ruang dan waktu dan tanpa pertemuan fisik [28]. Aplikasi ini digunakan karena selama pengembangan sistem, tim *Scrum* tidak dalam satu lokasi. Sehingga semua informasi bisa dengan mudah dan cepat diketahui oleh tim.



Gambar 5. Proses pengembangan SiLPBPA menggunakan aplikasi *Trello*

### 3.5 Menentukan *Product Backlog*

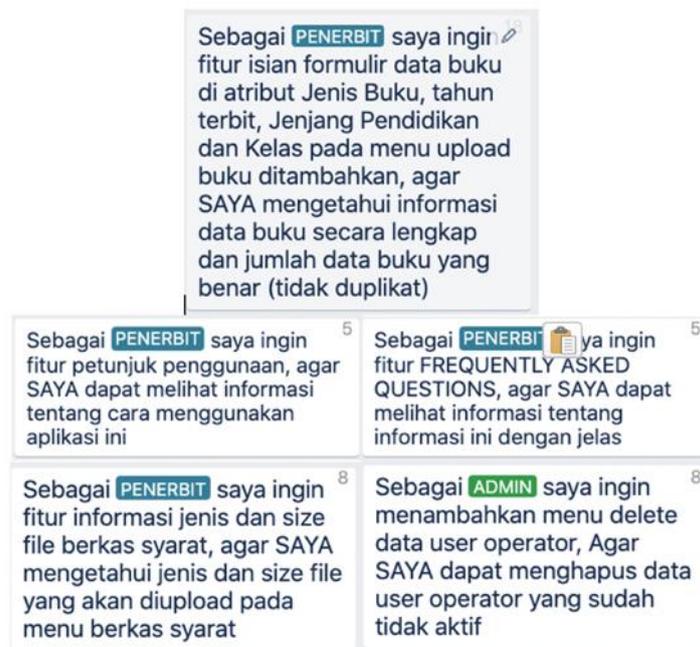
*Product Owner* (PO) membuat daftar, mengklasifikasi dan mengatur prioritas dari *Product Backlog Item* (PBI). Daftar PBI dibuat

atas hasil evaluasi pada Tabel 6. PO melakukan kolaborasi dengan *Development Team* (DT) dalam kegiatan ini. Berikut adalah hasilnya:

Tabel 7. *Product backlog*

No	Deskripsi Fitur yang akan Dikembangkan (PBI)	Prioritas	Estimasi Waktu (jam)
1	Menampilkan Menu Panduan Penggunaan pada halaman utama	Prioritas	12
2	Menampilkan Menu <i>Frequently Asked Question</i> (FAQ) pada halaman utama	Prioritas	12
3	Menambahkan isian kolom data buku pada atribut Jenis Buku, Tahun Terbit, Jenjang Pendidikan dan kelas sesuai dengan metadata buku dari penerbit sehingga identitas data buku lengkap dan tidak duplikat	High Prioritas	20
4	Menampilkan informasi jenis file dan <i>size</i> pada menu berkas syarat bagi <i>user</i> penerbit	Prioritas	12
5	Menambahkan menu <i>delete</i> data <i>user</i> operator pada <i>user admin</i>	Prioritas	14

Kemudian PBI dibuat menjadi *user story* untuk dimasukkan pada aplikasi *Trello*.



Gambar 6. *User Story* pada aplikasi *Trello*

### 3.6 Merencanakan *Sprint*

DT dan PO berkolaborasi dalam tahap ini. Tujuan dari tahap ini adalah membuat *Sprint Goal*. DT mengerjakan *Sprint Backlog* dalam

sebuah *sprint* dari PBI (diuraikan menjadi *task*) untuk mencapai *Sprint Goal*. Urutan *Sprint* yang sudah disepakati dan akan dijalankan oleh DT ada pada Tabel 8, 9,10, dan 11.

Tabel 8. *Sprint pertama menambahkan field atribut pada menu upload buku*

<i>Sprint Backlog</i>	<i>Sprint Goal</i>	<i>Task</i>	Estimasi Waktu (Jam)
Sebagai PENERBIT saya ingin fitur isian formulir data buku di atribut Jenis Buku, tahun terbit, Jenjang Pendidikan dan Kelas pada menu <i>upload</i> buku ditambahkan, agar SAYA mengetahui informasi data buku secara lengkap dan jumlah data buku yang benar (tidak duplikat)	OPERATOR Mendapatkan jumlah data buku yang sesuai dengan yang di- <i>upload</i> oleh <i>user</i> penerbit	<i>Create</i> fungsi <i>script</i> dan <i>table</i> atribut Jenis Buku	4
		<i>Create</i> fungsi <i>script</i> dan <i>table</i> atribut Jenis Buku atribut tahun terbit	4
		<i>Create</i> fungsi <i>script</i> dan <i>table</i> atribut Jenis Buku atribut Jenjang Pendidikan	4
		<i>Create</i> fungsi <i>script</i> dan <i>table</i> atribut Jenis Buku atribut Kelas	4
		Modifikasi Tampilan Menu <i>Upload</i> Buku	4
Total			20

Tabel 9. *Sprint kedua menambahkan menu panduan penggunaan dan frequently asked questions (FAQ)*

<i>Sprint Backlog</i>	<i>Sprint Goal</i>	<i>Task</i>	Estimasi Waktu (Jam)
1. Sebagai PENERBIT saya ingin fitur petunjuk penggunaan, agar SAYA dapat melihat informasi tentang cara menggunakan aplikasi ini. 2. Sebagai PENERBIT saya ingin fitur <i>FREQUENTLY ASKED QUESTIONS</i> , agar SAYA dapat melihat informasi tentang informasi ini dengan jelas	PENERBIT Dapat melihat informasi panduan penggunaan dan FAQ	<i>Create fungsi script menu Panduan Penggunaan</i>	4
		<i>Create table menu Panduan Penggunaan</i>	4
		<i>Create tampilan menu Panduan Penggunaan</i>	4
		<i>Create fungsi script menu FAQ</i>	4
		<i>Create table menu FAQ</i>	4
		<i>Create tampilan menu FAQ</i>	4
		<b>Total</b>	<b>24</b>

Tabel 10. *Sprint ketiga menambahkan informasi jenis dan size file pada menu upload berkas*

<i>Sprint Backlog</i>	<i>Sprint Goal</i>	<i>Task</i>	Estimasi Waktu (Jam)
Sebagai PENERBIT saya ingin fitur informasi jenis dan size file berkas syarat, agar SAYA mengetahui jenis dan size file yang akan di-upload pada menu berkas syarat	PENERBIT dapat melihat informasi jenis dan size file upload berkas syarat	<i>Create tampilan informasi jenis dan size file yang akan di-upload user</i>	6
		<i>Modifikasi script jenis dan size upload file</i>	6
<b>Total</b>		<b>12</b>	

Tabel 11. *Sprint keempat menambahkan menu delete data user operator pada user admin*

<i>Sprint Backlog</i>	<i>Sprint Goal</i>	<i>Task</i>	Estimasi Waktu (Jam)
Sebagai ADMIN saya ingin menambahkan menu <i>delete</i> data <i>user</i> operator, Agar SAYA dapat menghapus data <i>user</i> operator yang sudah tidak aktif	ADMIN dapat menghapus data dari operator	<i>Create script delete data</i>	5
		<i>Modifikasi table CRUD user operator</i>	5
		<i>Modifikasi tampilan</i>	4
<b>Total</b>		<b>14</b>	

### 3.7 Menjalankan *Sprint* dan *Daily Scrum*

DT mengerjakan beberapa *task* pada *Sprint* Pertama dari yang sudah disepakati pada *Sprint Backlog*. *Sprint* ini dilakukan selama 20 hari dengan pengerjaan 5 *task*. Waktu pengerjaan

yang diberikan untuk setiap *task*-nya adalah 4 jam. *Sprint* ini dilakukan selama 3 hari dari tanggal 6 April sampai dengan 8 April 2020. Berikut tampilan proses pengerjaan *Sprint* Pertama pada aplikasi *Trello*:



Gambar 7. Proses *sprint* pertama pada aplikasi *Trello*

DT menyelesaikan 1 jam lebih cepat dari *Sprint Pertama*. Berikut adalah *Product Increment* pada *Sprint Pertama*:

Berkas Syarat	
Judul Buku	<input type="text" value="Judul Buku"/>
Kategori	<input type="text" value="Fiqih"/>
Kelas	<input type="text" value="I"/>
Tahun	<input type="text" value="Tahun"/>
File	<input type="button" value="Choose File"/> No file chosen
<input type="button" value="Simpan"/> <input type="button" value="Histori Upload"/>	

Gambar 8. Kolom formulir menu *upload* buku sebelum dikembangkan pada sistem

Berkas Syarat	
Judul Buku	<input type="text" value="Judul Buku"/>
Jenis Buku	<input type="text" value="Guru"/>
Jenjang Pendidikan	<input type="text" value="Taman kanak-kanak"/>
Kategori	<input type="text" value="Fiqih"/>
Kelas	<input type="text" value="I"/>
Tahun	<input type="text" value="Tahun"/>
File	<input type="button" value="Choose File"/> No file chosen
<input type="button" value="Simpan"/> <input type="button" value="Histori Upload"/>	

Gambar 9. Kolom formulir menu *upload* buku setelah dikembangkan pada sistem

Pada *Sprint Kedua* DT mengerjakan 6 *task* yang sudah disepakati selama 24 jam. Target waktu pengerjaan dalam 1 *task* adalah 4 jam. Hari efektif pengerjaan *Sprint* ini adalah 3 hari.

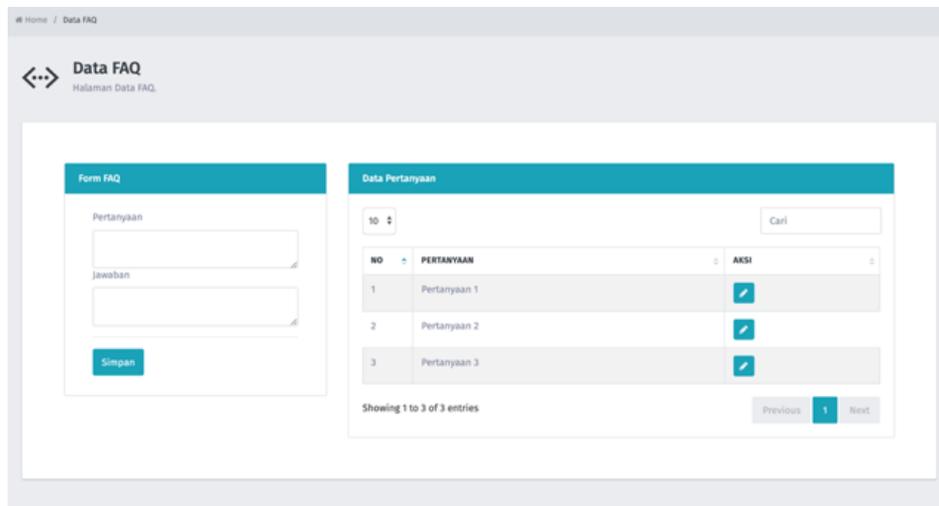
3 hari lainnya adalah hari libur. Dimulai dari tanggal 9 April sampai dengan 14 April 2020. Berikut adalah tampilan proses pengerjaan *Sprint Kedua* pada aplikasi *Trello*:



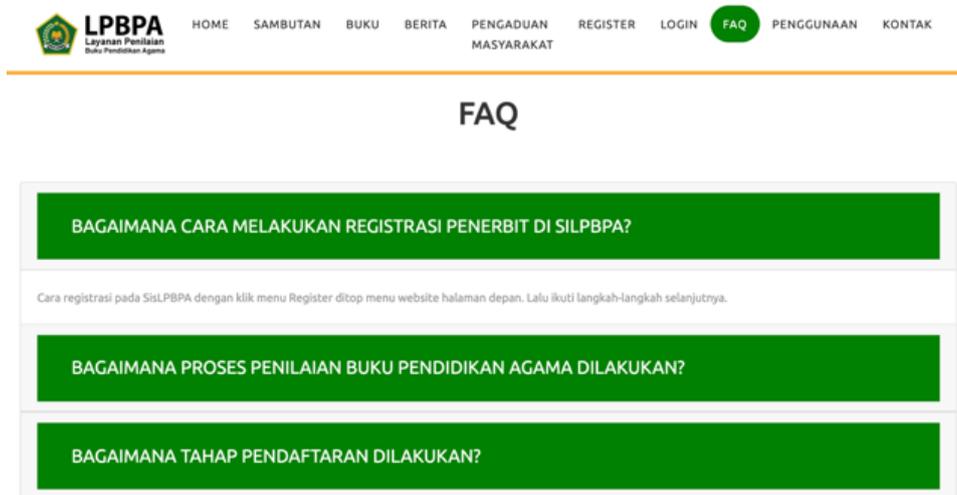
Gambar 10. Proses *sprint* kedua pada aplikasi Trello

Dalam menyelesaikan *Sprint* Kedua, DT melebihi target waktu yang sudah ditentukan yaitu lebih 2 jam. Dari waktu pengerjaan 24 jam jadi 26 jam. Namun *Sprint* Kedua sukses

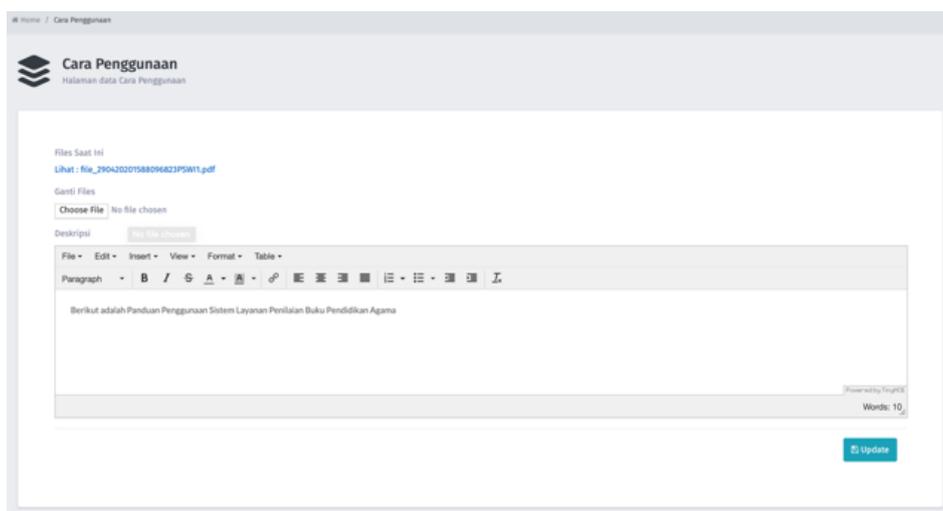
dijalankan dan diselesaikan dengan baik sehingga menghasilkan *Product Increment* sebagai berikut:



Gambar 11. Halaman *backend* hasil pengembangan menu FAQ



Gambar 12. Halaman *frontend* hasil pengembangan menu FAQ



Gambar 13. Halaman *backend* hasil pengembangan menu cara penggunaan



Gambar 14. Halaman *frontend* hasil pengembangan menu cara penggunaan



Gambar 15. Halaman *frontend* sebelum hasil pengembangan



Gambar 15. Halaman *frontend* sesudah hasil pengembangan

Pada *Sprint* Ketiga, DT melakukan proses pengerjaan selama 2 hari dari tanggal 15 April sampai dengan 16 April 2020. Dengan target

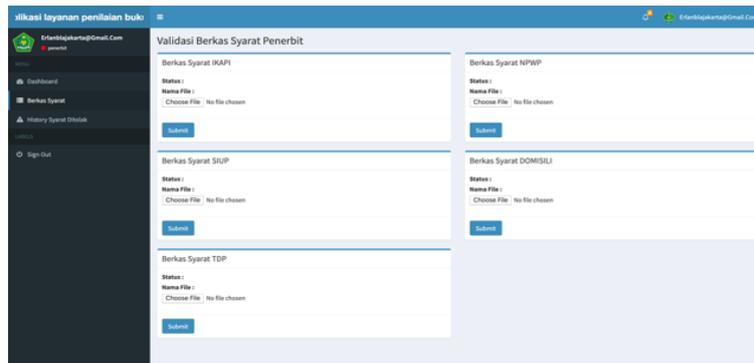
waktu 12 jam untuk 2 *task*. Berikut tampilan proses pengembangan sistem pada aplikasi Trello:



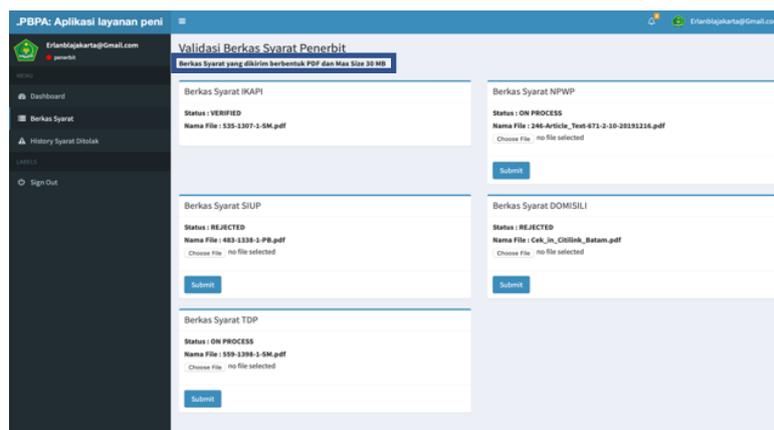
Gambar 16. *Sprint* ketiga pada aplikasi Trello

Pada *Sprint* ini, semua *task* diselesaikan lebih cepat 2 jam oleh DT dari target waktu yang telah ditentukan yaitu 10 jam, sehingga

menghasilkan *Product Increment* sebagai berikut:



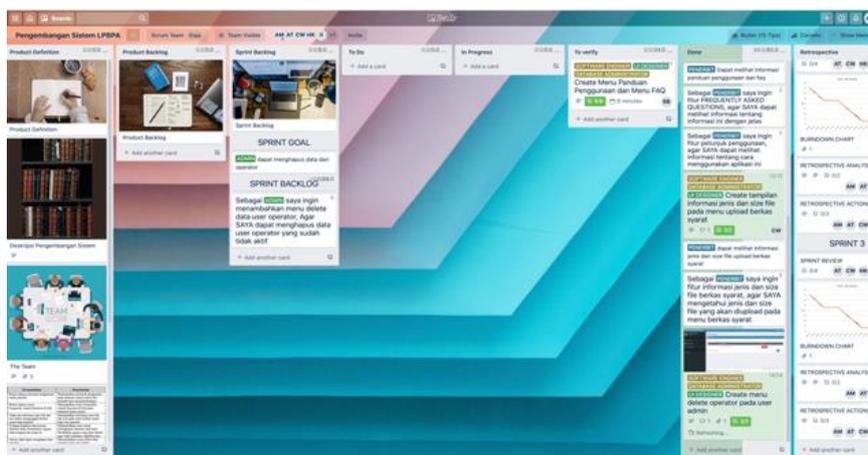
Gambar 17. Informasi jenis file dan size menu upload berkas syarat sebelum dikembangkan pada sistem



Gambar 18. Informasi jenis file dan size menu upload berkas syarat sesudah dikembangkan pada sistem

*Sprint* Keempat dilaksanakan dari tanggal 17 April sampai dengan 20 April 2020. Hari efektif kerja adalah 2 hari, sedangkan 2 harinya libur. *Sprint* ini merupakan *Sprint* terakhir dalam melakukan pengembangan SiLPBPA. Terdapat 3 *task* yang harus diselesaikan dengan

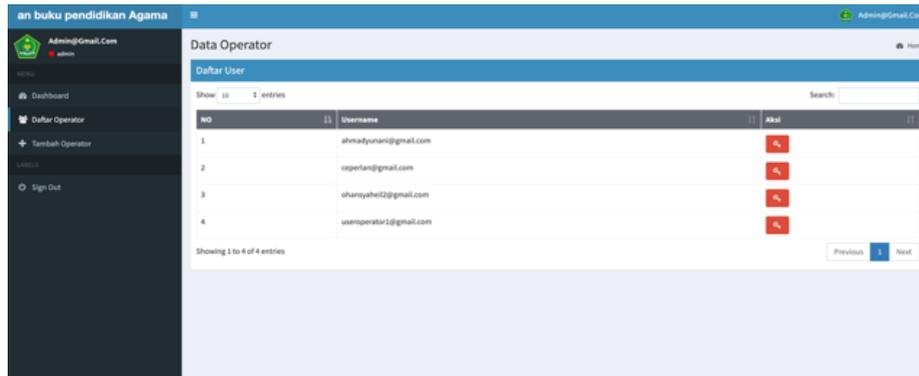
target waktu 14 jam. 2 *task* target waktunya 10 jam sedangkan sisanya 1 *task* target waktu yang ditentukan 4 jam. Berikut tampilan proses pengerjaan *Sprint* Keempat pada aplikasi *Trello*:



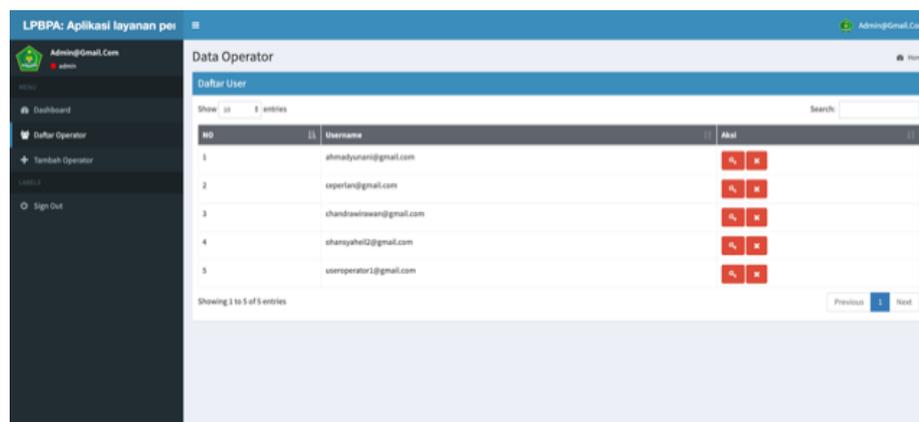
Gambar 19. *Sprint* keempat pada aplikasi *Trello*

DT menyelesaikan semua task lebih cepat 4 jam dari target waktu yang sudah ditentukan

yaitu 10 jam. Berikut adalah tampilan *Product Increment* yang dihasilkan pada tahap ini:



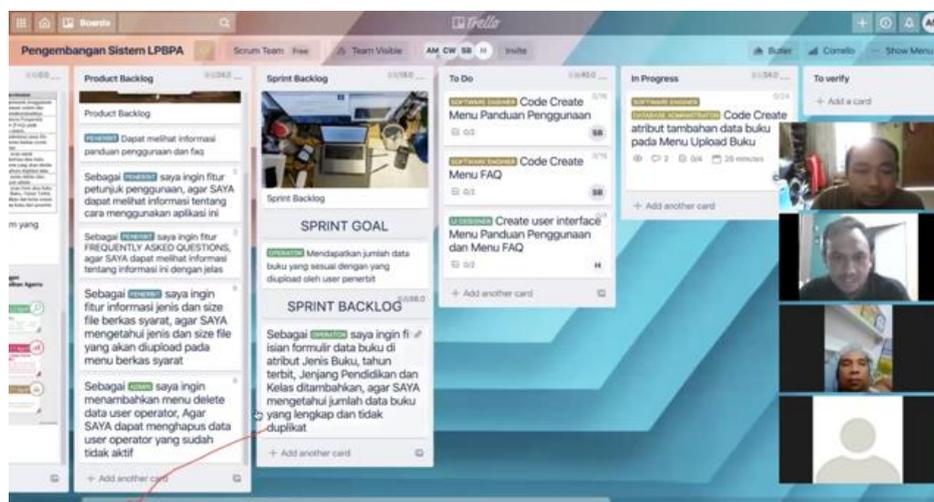
Gambar 20. Menu *delete operator* pada user admin sebelum dikembangkan



Gambar 20. Menu *delete operator* pada user admin sesudah dikembangkan

DT melakukan kegiatan *Daily Scrum* dengan durasi 15 menit selama *Sprint* Pertama sampai dengan *Sprint* Keempat berlangsung. Kegiatan ini dilakukan dengan pertemuan

secara daring. Tujuannya untuk membahas apa saja yang sudah dilakukan, kendala kemarin dan merencanakan aktifitas sekarang.

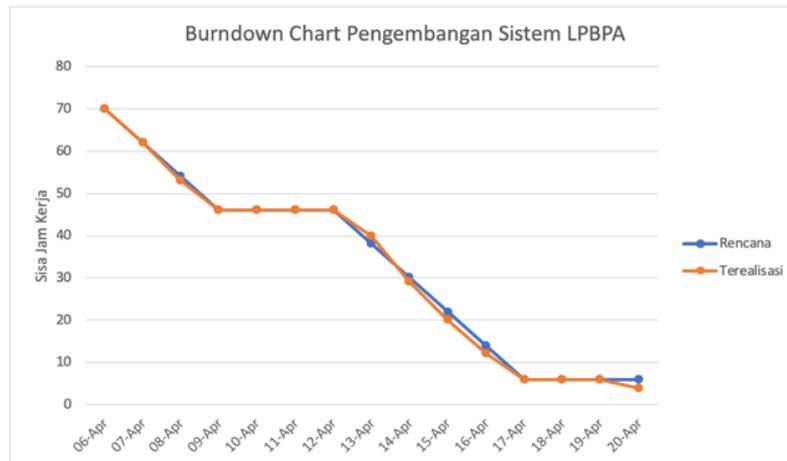


Gambar 21. Kegiatan *daily scrum development team* (DT) secara daring

### 3.8 Burndown Chart

*Burndown Chart* memudahkan tim *Scrum* dalam mendapatkan informasi untuk melakukan evaluasi pada akhir *sprint* [29]. Rencana dan realisasi pekerjaan yang sudah dilakukan dapat dilihat dengan *Burndown Chart*. Penyelesaian jumlah sisa jam kerja ditunjukkan oleh sumbu X (*Vertical*) sedangkan hari-hari dalam menjalankan *Sprint* ditunjukkan oleh sumbu Y (*Horizontal*). Pembaharuan

grafik dilakukan setiap hari pada pertemuan *Daily Scrum* oleh penggabungan estimasi sisa pekerjaan untuk semua *task* dalam *Sprint Backlog*. Garis biru pada *Burndown Chart* menunjukkan target waktu yang ditentukan sedangkan garis *orange* menunjukkan waktu yang terealisasi dari rencana yang ditentukan. Terlihat pada Gambar 22 bahwa jam kerja tidak sesuai dengan yang sudah ditentukan.

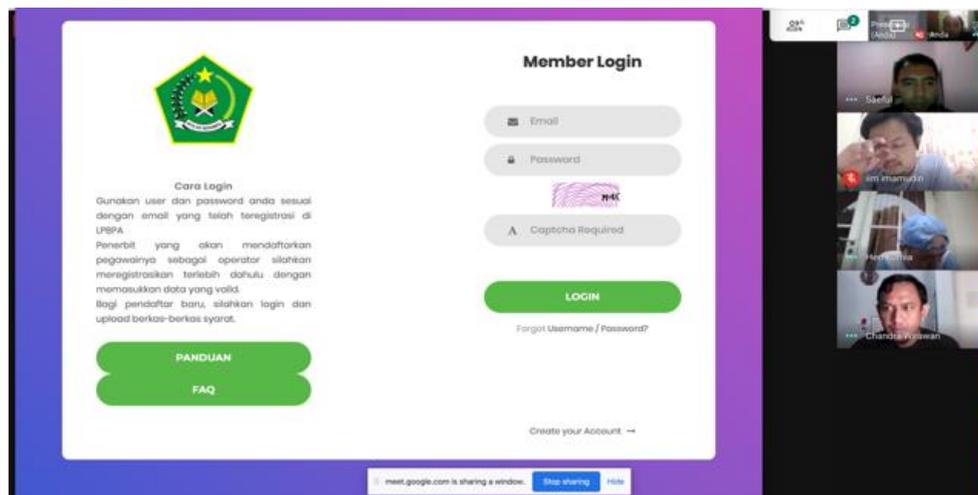


Gambar 22. *Burndown chart* pengembangan SiLPBPA

Tujuh puluh jam adalah target waktu untuk menyelesaikan pengembangan SiLPBPA ini. Dengan waktu efektif kerja adalah 10 hari dan 5 hari libur. Pada grafik, tanggal 13 April garis *orange* ada di atas garis biru karena penyelesaian pekerjaan lebih lambat 2 jam, sedangkan pada tanggal 8, 14, 15, 14 dan 20 April, Garis biru ada di bawah garis *orange* karena penyelesaian pekerjaan lebih cepat 7 jam. Jadi total waktu untuk menyelesaikan pengembangan SiLPBPA Versi 1.0 ini adalah 65 jam.

### 3.9 Melakukan *Sprint Review* dan *Sprint Retrospective*

*Sprint Review* dilakukan setelah *Product increment* dihasilkan dalam satu *Sprint*. Tujuan kegiatan ini adalah melakukan pemeriksaan dan penyesuaian terhadap fitur produk yang dihasilkan dari PBI. Tim *Scrum* beserta *Stakeholder* (pengguna sistem) dalam pengembangan SiLPBPA versi 1.0 terlibat semua dalam kegiatan ini. DT melakukan demonstrasi dari fitur produk yang dihasilkan. Semua peserta melakukan *review* dan diskusi mengenai fitur-fitur yang dikembangkan Semua peserta menyetujui fitur-fitur tersebut untuk segera di-*release*.



Gambar 23. Kegiatan *sprint review* secara daring

Kegiatan *Sprint Retrospective* dilaksanakan setelah *Sprint Review* yang dihadiri hanya oleh tim *Scrum*. Dalam pertemuan ini dibahas hambatan-hambatan yang terjadi selama satu *Sprint*. Hambatan yang terjadi ketika melakukan pengembangan SiLPBPA adalah lebih lambat 2 jam waktu pengerjaan pada *Sprint* Kedua ditanggal 13 April 2020. Hal ini terjadi karena adanya kesalahan komunikasi pada DT. Tim melakukan diskusi untuk menemukan solusi agar ke depannya tidak terjadi pada *sprint* selanjutnya.

#### IV. PENUTUP

Berdasarkan hasil penelitian yang telah dilakukan maka dapat disimpulkan hasil pengujian *usability* menggunakan SUS dengan 14 responden mendapatkan hasil skor SUS rata-rata 71. Ada pada Predikat C, Kategori *Acceptable*, dan dengan *Rating Good*. Artinya sistem ini hanya bisa digunakan untuk kegiatannya. Namun, ketika diimplementasikan SiLPBPA perlu dilakukan pengembangan. Hal ini dibuktikan dengan adanya beberapa saran dari pengguna pada hasil kuesioner. Saran dari *user* tersebut kemudian diidentifikasi permasalahan dan dibuat penyelesaiannya. Pengembangan sistem dilakukan dengan kerangka kerja *Scrum*. *Scrum* terbukti memiliki tingkat keberhasilan yang tinggi dalam pengembangan sistem ini. Hal ini terlihat pada *Burndown Chart*. Target waktu yang ditentukan adalah 70 jam dapat selesai hanya dalam waktu 65 jam. Kemudian *Product Increment* yang dihasilkan pada semua *Sprint* sudah memenuhi kebutuhan pengguna. Hal ini Terlihat pada

kegiatan *Sprint Review*, semua peserta (Tim *Scrum* dan pengguna sistem) menyetujui hasilnya. Sehingga SiLPBPA dapat diimplementasikan pada kegiatan layanan penilaian buku Pendidikan Agama selanjutnya. Namun, untuk mengetahui tingkat keberhasilan implementasi sistem yang sudah dikembangkan perlu dilakukan penelitian analisis kualitas *usability* kembali. Ke depannya SiLPBPA ini perlu dilakukan pengembangan lebih lanjut. Terutama untuk fitur pengelolaan *user* keseluruhan, Integrasi *Backend* dengan *Frontend*, Pengelolaan Data Buku Pendidikan Agama, pengelolaan laporan akhir buku layak dinilai dan buku layak diterbitkan dan proses penilaian BPA di dalam sistem.

#### DAFTAR PUSTAKA

- [1] Kementerian Agama Republik Indonesia, Peraturan Menteri Agama Republik Indonesia Nomor 9 Tahun 2018 Tentang Buku Pendidikan Agama. Indonesia, 2018, pp. 1–11.
- [2] Badan Litbang dan Diklat Kemenag Republik Indonesia, Keputusan Kepala Badan Penelitian Dan Pengembangan Dan Pendidikan Dan Pelatihan Kementerian Agama Republik Indonesia Nomor 51 Tahun 2018 Tentang Penulisan, Penilaian, Dan Penerbitan Buku Pendidikan Agama. Indonesia, 2019, pp. 1–32.
- [3] B. Pudjoatmodjo and R. Wijaya, “Tes kegunaan (*usability testing*) pada aplikasi kepegawaian dengan menggunakan *system usability scale*,” Semin. Nas.

- Teknol. Inf. dan Multimed. 2016, pp. 37–42.
- [4] D. Setiawan, S. L. Wicaksono, and N. Rafianto, “Evaluasi usability e-learning moodle dan google classroom menggunakan SUS quisionnare,” *JAMI J. Ahli Muda Indonesia.*, vol. 1, no. 1, Mei 2020, pp. 55–64.
- [5] A. W. Soejono, et al., “Evaluasi usability website unriyo menggunakan system usability scale (studi kasus : website Unriyo),” *J. Teknol. Inf.*, vol. 13, no. 1, 2018, pp. 29–37.
- [6] W. U. Martoyo and Falahah, “Kajian evaluasi usability dan utility pada situs web,” in *Seminar Nasional Sistem Informasi Indonesia*, 2015, no. 2-3, November 2015, pp. 537–544.
- [7] U. Ependi, T. B. Kurniawan, and F. Panjaitan, “System usability scale vs heuristic evaluation: a review,” *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 10, no. 1, 2019, pp. 65–74.
- [8] R. V. Anand and M. Dinakaran, “Popular agile methods in software development: review and analysis,” *Int. J. Sci. Tech. Adv.*, vol. 2, no. 4, 2016, pp. 147–150.
- [9] K. Schwaber and J. Sutherland, *Panduan Scrum*, no. November. 2017.
- [10] D. Murdiani, A. Yudhana, and S. Sunardi, “Implementasi agile method dalam pengembangan jurnal elektronik di lembaga penelitian non-pemerintahan (nGo),” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 4, p. 709, 2020.
- [11] W. Warkim, et al., “Penerapan metode scrum dalam pengembangan sistem informasi layanan kawasan,” *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 2, 2020, pp. 365–378.
- [12] H. Hutrianto and A. Putra, “Implementasi scrum model dalam pengembangan aplikasi pelaporan sampah sebagai wujud smart cleaning,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 5, no. 1, 2020, pp. 9–19.
- [13] R. A. Azdy and A. Sn, “Implementasi scrum pada pengembangan software terdistribusi,” in *Seminar Nasional Informatika 2012 (Semnas IF 2012) UPN “Veteran” Yogyakarta*, Volume 1 No 2 2012: Network and Security, p. B-32-B-37, 2012.
- [14] P. A. G. Permana, “Scrum method implementation in a software development project management,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 6, no. 9, 2015, pp. 198–204.
- [15] E. S. F. Cardozo, et al, “Scrum and productivity in software projects: A systematic literature review,” in *14th International Conference on Evaluation and Assessment in Software Engineering (EASE) (EASE)*, 2010, pp. 1–4.
- [16] H. Harjono and M. Hamka, “Implementasi framework scrum dalam pengembangan sistem informasi jabatan fungsional akademik,” in *Seminar Nasional Hasil - Hasil Penelitian Dan Pengabdian Pada Masyarakat III Tahun 2017 “Pengembangan sumber daya menuju masyarakat madani berkearifan lokal,”* 2018, December 2017, pp. 239–244.
- [17] J. Partogi, *Manajemen Modern dengan Scrum-Sebuah Petualangan Baru di Abad 21 Menjadi Manajer Software Development Modern*. Yogyakarta: Penerbit ANDI, 2015.
- [18] Z. Sharfina and H. B. Santoso, “An Indonesian adaptation of the System Usability Scale (SUS),” *2016 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACSIS 2016*, 2017, pp. 145–148.
- [19] H. N. I. Aprilia, P. I. Santoso, and R. Ferdiana, “Pengujian usability website menggunakan system usability scale,” *J. IPTEKKOM J. Ilmu Pengetah. Teknol. Inf.*, vol. 17, no. 1, 2015, pp. 31–38.
- [20] E. Susilo, “Cara menggunakan system usability scale (SUS) pada evaluasi usability,” 2019. [Online]. Available: <https://www.edisusilo.com/cara-menggunakan-system-usability-scale/>. [Accessed: 14-Mar-2020].
- [21] J. Brooke, *SUS: A “Quick and dirty” Usability Scale*, 1st ed. London: Taylor and Francis, 1996.
- [22] J. Brooke, “SUS: a retrospective,” *J. Usability Stud.*, vol. 8, no. 2, 2013, pp. 29–40.
- [23] J. Sauro, “Measuring usability with the System Usability Scale (SUS),” *Measuring U*, 2011. [Online]. Available: <https://measuringu.com/sus/>. [Accessed: 25-Sep-2019].
- [24] U. Efendi and Q. Widayati, “Extreme programming study method case study on designing of accounting term

- dictionary,” in *The Third International Conference On Engineering And Technology Development*, 2014, pp. 1–16.
- [25] A. Bangor, P. Kortum, and J. Miller, “Determining what individual SUS scores mean: adding an adjective rating scale,” *J. Usability Stud.*, vol. 4, no. 3, 2009, pp. 114–123.
- [26] K. S. Rubin, “Essential scrum: A practical guide to the most popular agile process,” 2nd ed. New Jersey: Pearson Education, 2013.
- [27] J. Sauro, “Does better usability increase customer loyalty?,” in *Measuring U*, 2010. [Online]. Available: <https://measuringu.com/usability-loyalty/>. [Accessed: 20-Mar-2020].
- [28] L. Moon, “The top trello tips for beginners to boost productivity,” 2020.
- [29] M. A. Firdaus, “Implementasi kerangka kerja scrum pada manajemen pengembangan sistem informasi,” *s*, vol. 1, no. 2, 2017, pp. 283–288.

### **Hak Cipta**

Semua naskah yang tidak diterbitkan, dapat dikirimkan di tempat lain. Penulis bertanggung jawab atas ijin publikasi atau pengakuan gambar, tabel dan bilangan dalam naskah yang dikirimkannya. Naskah bukanlah naskah jiplakan dan tidak melanggar hak-hak lain dari pihak ketiga. Penulis setuju bahwa keputusan untuk menerbitkan atau tidak menerbitkan naskah dalam jurnal yang dikirimkan penulis, adalah sepenuhnya hak Pengelola. Sebelum penerimaan terakhir naskah, penulis diharuskan menegaskan secara tertulis, bahwa tulisan yang dikirimkan merupakan hak cipta penulis dan menegaskan hak cipta ini pada pengelola.