



## Analisis Aplikasi Iuran Pengelolaan Lingkungan Berbasis Web Dengan Proses Monitoring dan Evaluasi COBIT 4.1 (Studi Kasus Perumahan Metro Residence)

\* Dionisius Riyan Edlianto <sup>1</sup>, Agung Budi Susanto <sup>2</sup>, Tukiyat <sup>3</sup>

<sup>1,2,3</sup>Teknik Informatika, Program Pascasarjana, Universitas Pamulang, Tangerang Selatan, Banten

Email: <sup>1</sup> dionisius.riyan@gmail.com, <sup>2</sup> dosen02680@unpam.ac.id, <sup>3</sup> dosen02711@unpam.ac.id

### ABSTRACT

*Metro Residence Housing has a total of 381 houses. The sales admin manually inputs environmental management contribution data into Excel every month. To address this, an application was created using the Scrum method, chosen for its adaptability and efficiency. Sprints one to three, each lasting two weeks, completed in six weeks. Post-development, a black box test with eight menu tests confirmed the system's functionality, with all tests passed. Subsequent direct testing with the admin over a month led to application acceptance. An IT governance audit followed, involving five respondents: sales admin, estate admin, admin manager, branch IT staff, and HO IT staff. The COBIT 4.1 audit rated Bogor Metro Residence at level three, indicating standardized, documented, and communicated IT procedures. Limited IT staff understanding prevented a higher score. The application can be further developed into a mobile app for residents to monitor bills and make payments.*

*Keywords: Environmental Management Fees, COBIT 4.1, Scrum, Audit, Blackbox Testing*

### ABSTRAK

Perumahan Metro Residence memiliki 381 rumah. Admin penjualan memasukkan data kontribusi pengelolaan lingkungan secara manual ke dalam Excel setiap bulan. Untuk mengatasi masalah ini, dibuat aplikasi dengan metode Scrum yang dipilih karena adaptabilitas dan efisiensinya. Sprint satu hingga tiga, masing-masing berdurasi dua minggu, selesai dalam enam minggu. Setelah pengembangan, pengujian black box dengan delapan uji menu memastikan fungsionalitas sistem, dengan semua uji lulus. Pengujian langsung dengan admin selama sebulan menghasilkan penerimaan aplikasi. Audit tata kelola IT melibatkan lima responden: admin penjualan, admin estate, manajer admin, staf IT cabang, dan staf IT kantor pusat. Audit COBIT 4.1 menilai Perumahan Metro Residence Bogor pada level tiga, menunjukkan prosedur IT yang distandarisasi, didokumentasikan, dan dikomunikasikan. Pemahaman terbatas staf IT mencegah skor lebih tinggi. Aplikasi ini dapat dikembangkan menjadi aplikasi mobile untuk memudahkan warga memantau tagihan dan melakukan pembayaran.

Kata Kunci : Iuran Pengelolaan Lingkungan, COBIT 4.1, Scrum, Audit, Pengujian *Blackbox*.

## 1. PENDAHULUAN

Iuran bulanan merupakan kesepakatan bersama seluruh warga dimana iuran tersebut dialokasikan untuk kepentingan seperti pengelolaan pembersihan sampah, pembayaran keamanan, dan untuk kepentingan lainnya [1], [2]. Iuran yang dibayarkan ini sifatnya wajib dibayarkan setiap satu bulan sekali. Setiap blok pada perumahan dan tipe perumahan memiliki IPL yang berbeda, untuk saat ini pada tahun 2022 terdapat 3

*cluster* pada perumahan *Metro Residence*. Pencatatan ini dilakukan dimulai dari bulan Mei 2022, dimana pemberlakuan pencatatan *manual* ini guna untuk kepentingan perpajakan. Pencatatan ini masih menggunakan aplikasi *Excel* yang diketik *manual* dalam proses pembuatan kwitansi dan laporan bulanannya.

Tabel 1. Data Transaksi IPL Bulan Mei – Oktober 2022

No	Jumlah Transaksi PPN	Jumlah Transaksi Non-PPN	Bulan/Tahun
1	380	220	05/2022
2	275	189	06/2022
3	311	162	07/2022
4	332	183	08/2022
5	290	176	09/2022
6	298	182	10/2022

Kesulitan yang timbul saat ini adalah *sales admin* kewalahan untuk melakukan penginputan manual, karena hanya ada *sales admin* dan data yang didapat sangat banyak, seperti yang terlihat pada Tabel 1. Hal ini menyebabkan kesalahan input yang mungkin terjadi selama proses penginputan dari rekening koran menuju *Excel*. Kesalahan tersebut dapat berupa salah nominal pembayaran, salah penulisan nama penghuni, kesalahan tanggal pembuatan kwitansi dan lain sebagainya. Banyaknya transaksi membutuhkan kecepatan dan ketepatan, yang membuat perlu adanya aplikasi yang dapat mempermudah pekerjaan *sales admin*, sehingga tenggat waktu yang diberikan bagian *finance* dapat terpenuhi.

Aplikasi yang dibuat nanti merupakan aplikasi berbasis *website* menggunakan *framework* *Laravel* dan bersifat *online* [3], sehingga *Sales admin* dapat mengerjakan inputan dari manapun. Inputan ini nantinya akan dapat di-*import* ke *Excel* untuk bahan lampiran *print out* rekapan kwitansi untuk bagian *finance* yang mana transaksi IPL ini harus dimasukkan ke dalam *program* ERP perusahaan. Dengan adanya aplikasi ini, pekerjaan *Sales admin* akan menjadi lebih efisien dan kesalahan yang terjadi dapat diminimalisir. Setelah aplikasi ini selesai, akan dilakukan monitoring dan evaluasi agar dapat berjalan sesuai standar COBIT 4.1 dan memenuhi standar sistem secara umum.

## 2. METODE

### 2.1. Pengumpulan data

#### 2.1.1. Kuesioner

Kuesioner ini dibuat berdasarkan kerangka kerja COBIT 4.1 dengan menggunakan *domain* monitor dan evaluasi. Kuesioner ini akan diberikan saat

pengujian sistem sedang berjalan dan disebarikan kepada 5 Responden yaitu, kepada dua *sales admin*, satu manajer *sales admin*, satu *staff* IT dan Karyawan *Estate*.

### 2.1.2. Teknik Observasi

Dalam penelitian ini juga dilakukan *observasi* yang berhubungan langsung dengan *staff* yang menggunakan sistem penginputan transaksi IPL pada Perumahan Metro Residence.

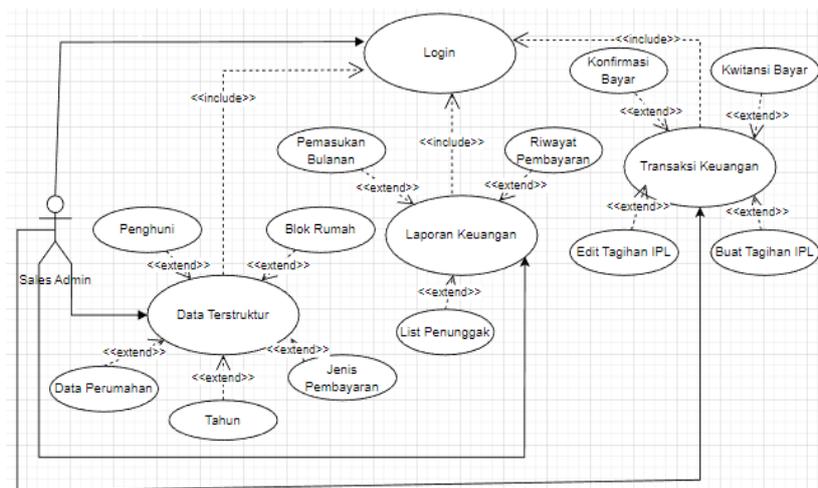
### 2.1.3. Wawancara

Dalam penelitian ini memakai teknik wawancara, wawancara dilakukan satu *sales admin*, satu manajer *sales admin*.

### 2.1.4. Studi Pustaka

Metode yang dilakukan adalah dengan cara mencari bahan yang mendukung dalam pendefinisian masalah melalui buku-buku, internet, yang erat kaitannya dengan objek permasalahan.

## 2.2. Use Case Perancangan Aplikasi



Gambar 1. Use Case Diagram IPL

Dari *use case diagram* di atas, dapat dilihat bahwa terdapat dua *user* dengan *Menu* yang telah dijabarkan di atas. *Sales admin* dan bagian *finance* harus mengakses *login* terlebih dahulu sebelum dapat mengakses *Menu* atau sub *Menu* dalam sistem tersebut. Untuk *Sales admin* dapat mengakses *Menu* data terstruktur yang didalamnya terdapat sub *Menu* penghuni, data perumahan, tahun, jenis pembayaran dan blok rumah, *Menu* laporan keuangan terdapat sub *Menu* pemasukan bulanan, riwayat pembayaran dan *list* penunggak. *Menu* transaksi keuangan terdapat sub *Menu* konfirmasi bayar, kwitansi bayar, *edit* tagihan IPL dan *buat* tagihan IPL.

### 2.3. Metode Scrum

Menurut Schwaber & Sutherland, Scrum adalah sebuah kerangka kerja yang dapat mengatasi suatu masalah kompleks yang selalu berubah, dan juga dinilai dapat memberikan kualitas produk yang baik sesuai dengan keinginan pengguna secara kreatif dan produktif [4]. Scrum terdiri dari Team Scrum dengan peran masing-masing, serta acara Scrum (Scrum event), artefak Scrum (Scrum artifact), dan aturan main Scrum. Komponen-komponen Scrum memiliki tujuan dan peran dalam mencapai keberhasilan menggunakan Scrum [5].

Tabel 2. Daily Sprint 1.

ID	Item Backlog	Story	Task	Assignment	Est(hari)
1	Perancangan UML		Membuat analisa kebutuhan fungsional sistem dan flowchart	DRE	1
			Membuat use case diagram	DRE	2
			Membuat activity diagram	DRE	2
			membuat class diagram	DRE	2
2	Membuat Login Admin	Sebagai Sales admin saya dapat login ke sistem	membuat skema database user	DRE	0.5
			membuat desain UI login Admin	DRE	0.5
			implementasi desain UI login admin ke koding	DRE	1
			melakukan test fitur login	DRE	1
3	Membuat Dashboard home	Sebagai sales admin saya dapat melihat menu dan dashboard aplikasi website	Membuat desain UI dashboard	DRE	0.5
			Membuat UI desain sidebar, footer dan header	DRE	1
			Membuat koding tampil total data	DRE	1
			Membuat skema database Penghuni, IPL, tahun bulan.	DRE	0.5
4	Membuat sub menu penghuni	Saya sebagai sales admin dapat mengakses sub menu Penghuni dan melakukan penginputan dengan sukses	Membuat UI desain sub menu penghuni	DRE	1
			Membuat form UI sub menu penghuni	DRE	1
			Implementasi desain UI sub menu penghuni ke koding	DRE	2
			melakukan test fitur penghuni	DRE	1
5	Membuat sub menu perumahan	Saya sebagai sales admin dapat mengakses sub menu Perumahan dan melakukan penginputan dengan sukses	Membuat UI Desain sub menu perumahan	DRE	0.5
			membuat form UI sub menu perumahan	DRE	0.5
			Implementasi desain UI Sub menu perumahan ke koding	DRE	1
			melakukan test fitur perumahan	DRE	1

Pada Tabel 2 menjelaskan task yang dikerjakan pada daily sprint 1. Mulai dari Perancangan UML, membuat login Admin, Membuat Dashboard Home, Membuat Sub Menu Penghuni dan membuat sub menu perumahan.

Tabel 3. Daily Sprint 2

ID	Item Backlog	Story	Task	Assignment	Est(hari)
6	Membuat sub menu Tahun	Saya sebagai sales admin dapat mengakses sub menu Tahun dan melakukan penginputan dengan sukses	Membuat UI Desain sub menu tahun	DRE	0.5
			membuat form UI sub menu tahun	DRE	0.5
			Implementasi desain UI sub menu tahun ke koding	DRE	1
			melakukan test fitur tahun	DRE	1
7	Membuat sub menu jenis pembayaran	Saya sebagai sales admin dapat mengakses sub menu jenis pembayaran dan melakukan penginputan dengan sukses	Membuat UI desain sub menu jenis pembayaran	DRE	0.5
			membuat form UI sub menu jenis pembayaran	DRE	0.5
			implementasi desain UI sub menu jenis pembayaran ke koding	DRE	1
			melakukan test fitur jenis pembayaran	DRE	1
8	Membuat sub menu Blok Perumahan	Saya sebagai sales admin dapat mengakses sub menu blok dan melakukan penginputan dengan sukses	membuat UI desain sub menu blok	DRE	0.5
			membuat form UI sub menu blok	DRE	0.5
			implementasi desain UI sub menu blok ke koding	DRE	1
			melakukan test fitur blok	DRE	1
9	Membuat sub menu Konfirmasi bayar	Saya sebagai sales admin dapat mengakses sub menu konfirmasi bayar dan melakukan penginputan dengan sukses	membuat UI desain sub menu konfirmasi bayar	DRE	0.5
			membuat form UI sub menu konfirmasi bayar	DRE	0.5
			implementasi desain UI sub menu konfirmasi bayar ke koding	DRE	3
			melakukan test fitur konfirmasi bayar	DRE	1
10	Membuat sub menu Kwitansi Bayar	Saya sebagai sales admin dapat mengakses sub menu Kwitansi Bayar dan melakukan penginputan serta print dengan sukses	membuat UI desain sub menu kwitansi bayar	DRE	0.5
			membuat form UI print sub menu kwitansi bayar	DRE	0.5
			implementasi desain UI sub menu kwitansi bayar ke koding	DRE	3
			melakukan test fitur kwitansi bayar	DRE	1

Pada *Table 3* menjelaskan *task* yang dikerjakan pada *daily sprint 2*. Mulai dai Membuat sub menu tahun, membuat sub menu jenis pembayaran, membuat menu blok Perumahan, membuat menu konfirmasi bayar dan kwitansi bayar.

Tabel 4. *Daily Sprint 3*

ID	Item Backlog	Story	Task	Assignment	Est(hari)
11	Membuat sub menu buat tagihan IPL	Saya sebagai sales admin dapat mengakses sub menu buat tagihan IPL dan melakukan penginputan dengan sukses	membuat UI desain sub menu buat tagihan IPL	DRE	0.5
			membuat form UI sub tagihan IPL	DRE	0.5
			implementasi desain sub menu tagihan IPL ke koding	DRE	1
			melakukan test fitur sub menu tagihan IPL	DRE	1
12	Membuat sub menu Edit tagihan IPL	Saya sebagai sales admin dapat mengakses sub menu edit tagihan IPL dan melakukan penginputan dengan sukses	membuat UI desain sub menu edit tagihan IPL	DRE	0.5
			membuat form UI sub menu edit tagihan IPL	DRE	0.5
			implementasi desain sub menu edit tagihan IPL ke koding	DRE	1
			melakukan test fitur edit tagihan IPL	DRE	1
13	Membuat sub menu Pemasukan Tahunan	Saya sebagai sales admin dapat mengakses sub menu Pemasukan Tahunan	membuat UI desain sub menu pemasukan tahunan	DRE	0.5
			implementasi desain sub menu pemasukan tahunan ke koding	DRE	1.5
			melakukan test fitur pemasukan tahunan	DRE	1
14	Membuat sub menu sub list penunggak	Saya sebagai sales admin dapat mengakses sub menu list penunggak	membuat UI desain sub menu list penunggak	DRE	0.5
			implementasi desain sub menu list penunggak ke koding	DRE	0.5
			melakukan test fitur sub list penunggak	DRE	1
15	Membuat sub menu riwayat pembayaran	Saya sebagai sales admin dapat mengakses sub menu riwayat pembayaran	membuat UI desain sub menu riwayat pembayaran	DRE	0.5
			implementasi desain sub menu riwayat pembayaran ke koding	DRE	0.5
			melakukan test fitur sub menu riwayat pembayaran	DRE	1
16	Membuat sub menu tambah user	Saya sebagai sales admin dapat mengakses sub menu tambah user	membuat UI desain sub menu tambah user	DRE	0.5
			implementasi desain sub menu tambah user ke koding	DRE	3
			melakukan test fitur sub menu tambah user	DRE	0.5

Pada *Tabel 4* menjelaskan *task* yang dikerjakan pada *daily sprint 3*. mulai dari membuat menu tagihan IPL, membuat menu edit tagihan ipl, pemasukanan tahunan, list penunggak, riwayat pembayaran dan tambah user.

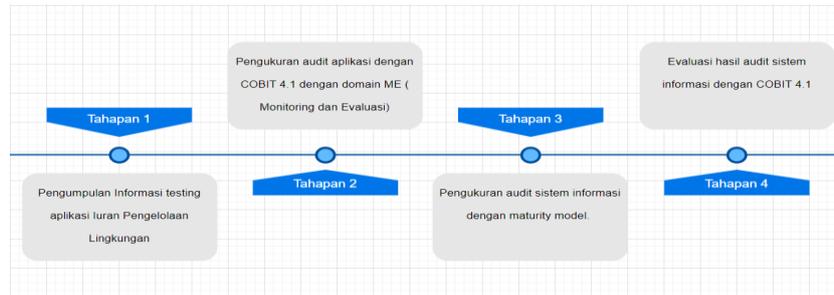
#### 2.4. *BlackBox Testing*.

*BlackBox Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya *field* data *entri* yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang *valid* [6].

#### 2.5. COBIT 4.1

Menurut Andry dalam buku [7] menjelaskan bahwa COBIT adalah merupakan *A set of best practice (framework)* bagi pengelola teknologi *inform asi (IT management)*. COBIT disusun oleh *IT Governance Institute (ITGI)* dan *Information Systems Audit and Information Systems Audit and Control Association (ISACA)*, Tepatnya *Information Systems Audit and Control Foundation's (ISACA)* pada tahun 1992. Edisi pertamanya dipublikasikan pada tahun 1996, edisi kedua pada tahun 1998, edisi ketiga

tahun 2000 (*versi on-line* dikeluarkan tahun 2003), edisi keempat pada Desember 2005 dan saat ini adalah edisi kelima pada 2012 [8].

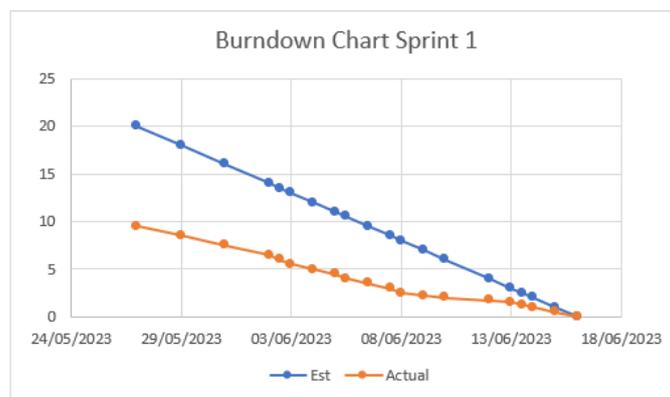


Gambar 2. Tahapan audit COBIT 4.1 Domain Monitoring dan Evaluasi.

### 3. HASIL DAN PEMBAHASAN

Pada bab ini akan dilakukan Analisis dan pengujian sistem , penulis menggunakan metode pengembangan sistem *Scrum* terhadap aplikasi Iuran Pengelolaan Lingkungan pada Perumahan *Metro Residence* yang sedang dibangun. Tahapan ini dilakukan setelah percangan sistem selesai dilakukan. Selanjutnya sistem akan di *testing* langsung dengan *user* untuk menguji hasil sistem yang telah dibuat, *testing* yang digunakan sendiri menggunakan metode *black box testing* dan akan dilihat kelayakan serta akan dilakukan perbaikan dan akan diuji kembali sampai aplikasi dapat digunakan oleh *user* [9]. *indicator* bahwa sistem tersebut dapat diterima dan dapat dikatakan baik. Setelah sistem dikatakan tidak ada masalah dan baik, maka selanjutnya dilakukan audit untuk menilai kelayakan tata kelola informasi pada perusahaan tersebut, hal ini berkaitan karena sebuah sistem yang baik diperlukan juga sumber daya manusia yang mengetahui dan menyadari pentingnya tata kelola IT untuk menunjang perkembangan perusahaan dan sistem itu sendiri.

#### 3.1. Daily Sprint Scrum

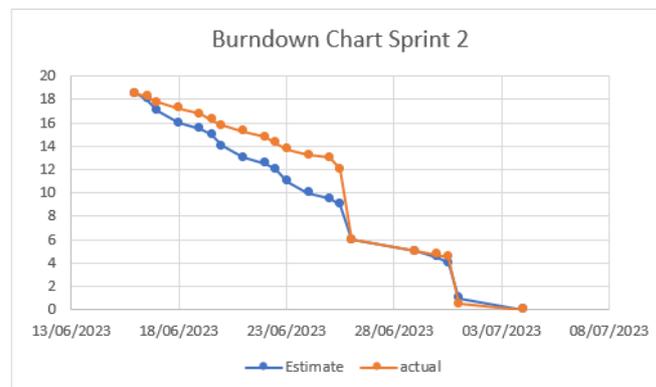


Gambar 3. Burndown Chart Sprint 1

Gambar 3 menunjukkan 5 item *backlog* dan 21 estimasi satuan (hari) atau disebut *work remaining* yang didapat dari perencanaan :

1. Tujuan *sprint* = Perancangan aplikasi dan pembuatan UML, *Login Admin*, *Dashboard Home*, *Sub Menu Penghuni*, *Membuat Sub Menu Perumahan*.
2. Panjang *sprint* = 15 hari → *Velocity* (Perkiraan yang dapat dikerjakan dalam)
3.  $Man\ days = 1(\text{orang}) \times 15(\text{hari}) = 15$
4.  $Focus\ factor = 71\% \ (Velocity / work\ remaining : 15/21 * 100\%)$
5. Perkiraan kecepatan = 21 (estimasi satuan) x 71% (FF) = 14.91 Hari

Pada hari pertama *sprint*, tanggal 24 Mei 2023, tim memperkirakan bahwa ada 21 estimasi *work remaining* yang perlu diselesaikan berdasarkan perhitungan kecepatan tim. Namun agar tidak menyimpang dari tujuan *sprint* yang telah ditentukan, *Velocity* atau kecepatan yang dimasukkan 15 poin estimasi. Pada tanggal 16 Juni 2023, grafik Menunjukkan bahwa tidak ada estimasi *work remaining* yang tersisa, itu artinya tim dapat menyelesaikan semua *sprint backlog* dengan baik.

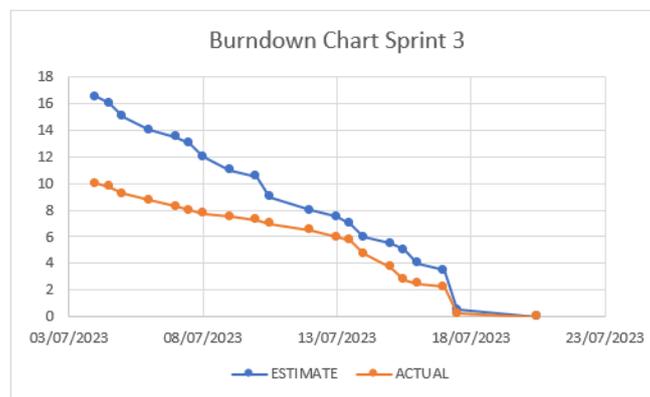


Gambar 4. *Burndown Chart Sprint 2*

Gambar 4 menunjukkan 5 item *backlog* dan 19 estimasi satuan (hari) atau disebut *work remaining* yang didapat dari perencanaan :

1. Tujuan *sprint* = Perancangan aplikasi dan pembuatan *Menu Tahun*, *Jenis pembayaran*, *Blok Perumahan*, *Konfirmasi Bayar* dan *Kwitansi Bayar*.
2. Panjang *sprint* = 14 hari → *Velocity* (Perkiraan yang dapat dikerjakan)
3.  $Man\ days = 1(\text{orang}) \times 14(\text{hari}) = 14$
4.  $Focus\ factor = 73\% \ (Velocity / work\ remaining : 14/19 * 100\%)$
5. Perkiraan kecepatan = 19 (estimasi satuan) x 73% (FF) = 13,87 Hari. Maka *product backlog* yang dimasukkan pada *sprint 2* sebanyak 14 poin estimasi.

Pada hari pertama *sprint*, tanggal 16 Juni 2023, tim memperkirakan bahwa ada 19 estimasi *work remaining* yang perlu diselesaikan berdasarkan perhitungan kecepatan tim. Namun agar tidak menyimpang dari tujuan *sprint* yang telah ditentukan, *Velocity* atau kecepatan yang dimasukkan 14 poin estimasi. Pada Tanggal 01 Juli 2023 dan 26 Juni 2023, terdapat Gap signifikan perbedaan estimasi dan *actual*, hal ini dikarenakan dalam proses pembuatan *website* ANALISIS pada koding mengalami kendala, sehingga memerlukan waktu lebih. Namun, Pada tanggal 04 Juli 2023, grafik Menunjukkan bahwa tidak ada estimasi *work remaining* yang tersisa, itu artinya tim dapat menyelesaikan semua *sprint backlog* dengan baik, namun melebihi dari Perkiraan Kecepatan. Hal ini berarti memerlukan tambahan orang pada pengembangan *program*.



Gambar 5. Burndown Chart Sprint 3

Gambar 5 menunjukkan 6 item *backlog* dan 17 estimasi satuan (hari) atau disebut *work remaining* yang didapat dari perencanaan :

1. Tujuan *sprint* = Perancangan aplikasi dan pembuatan *Menu* Buat Tagihan IPL, *Edit* Tagihan IPL, Pemasukan Tahunan, *List* Penunggak, Riwayat Pembayaran, *Tambah User*.
2. Panjang *sprint* = 12 hari → *Velocity* (Perkiraan yang dapat dikerjakan)
3.  $Man\ days = 1(\text{orang}) \times 10(\text{hari}) = 12$
4.  $Focus\ factor = 70\%$  (  $Velocity / work\ remaining : 12/17 * 100\%$  )
5. Perkiraan kecepatan = 17 (estimasi satuan) x 70% (FF) = 11.90 Hari. Maka *product backlog* yang dimasukkan pada *sprint* 2 sebanyak kurang lebih 12 poin estimasi.

Pada Gambar 5 Menunjukkan hari pertama *sprint*, tanggal 04 Juli 2023, tim memperkirakan bahwa ada sekitar 17 estimasi *work remaining* yang perlu diselesaikan

berdasarkan perhitungan kecepatan tim. Namun agar tidak menyimpang dari tujuan *sprint* yang telah ditentukan, *Velocity* atau kecepatan yang dimasukkan 12 poin estimasi. Pada tanggal 20 Juli 2023, grafik *Menunjukkan* bahwa tidak ada estimasi *work remaining* yang tersisa, itu artinya tim dapat menyelesaikan semua *sprint backlog* dengan baik.

### 3.2. Hasil BlackBox Testing

Tabel 5. Rekap Blackbox Testing

No	Menu	Pengamatan	Kesimpulan
1	Pengujian Halaman <i>Login</i>	Semua Tombol dapat berfungsi, dan <i>login</i> menggunakan <i>user</i> dan <i>password</i> sesuai <i>database</i>	Diterima
2	Pengujian <i>Menu</i> Penghuni	Semua tombol dapat dgiunakan dan tidak ada <i>error</i> ataupun kendala saat penginputan penghuni baru	Diterima
3	Pengujian Halaman <i>Menu Data</i> Perumahan	Dapat menambahkan data perumahan dalam <i>database</i> dan dapat melakukan <i>edit data</i> perumahan	Diterima
4	Pengujian <i>Menu</i> Tahun	Dapat menambahkan tahun pada <i>database</i> , dan menghapus tahun dalam <i>database</i> apabila ada kesalahan, dan tidak ada <i>error</i> yang muncul saat proses <i>input</i> ataupun hapus data.	Diterima
5	Pengujian <i>Menu</i> Jenis Pembayaran	Semua tombol dapat dgiunakan dan tidak ada <i>error</i> ataupun kendala saat penginputan jenis pembayaran.	Diterima
6	Pengujian <i>Menu</i> Blok	Semua tombol dapat dgiunakan dan tidak ada <i>error</i> ataupun kendala saat penginputan blok perumahan.	Diterima
7	Pengujian <i>Menu</i> Konfirmasi Bayar	Semua tombol dapat dgiunakan dan tidak ada <i>error</i> ataupun kendala saat penginputan konfirmasi bayar.	Diterima
8	Pengujian <i>Menu</i> Kwitansi Bayar	Semua tombol dapat dgiunakan dan tidak ada <i>error</i> ataupun kendala saat proses menampilkan <i>detail</i> kwitansi, serta dapat menampilkan data kwitansi secara benar.	Diterima

Berdasarkan Tabel 5, dijelaskan bahwa dari rekapan pengujian terhadap delapan menu pada sistem iuran pengelolaan lingkungan dijelaskan bahwa semua dapat diterima dengan sempurna, atau 100% baik. Hal ini berarti sistem sudah dapat dikatakan berjalan dengan baik dan dapat diimplementasikan dan digunakan oleh *user*.

### 3.3. Perhitungan Maturity Level COBIT 4.1

Setelah aplikasi selesai dibuat dan di uji coba selama satu bulan, maka langkah selanjutnya adalah melakukan proses *audit* pada aplikasi tersebut. Proses *audit* ini menggunakan *audit domain* Monitoring dan Evaluasi pada COBIT 4.1 dengan 5

Responden yang telah mengisi pada kuesioner lewat *Google Form* . Dari hasil pengolahan data *Google Form* maka didapat hasil *compliance* tiap *statement*. *Compliance* ini nanti akan diolah untuk mendapatkan *Compliance Value*, Setelah didapat *compliance value* maka akan dinormalisasikan dengan total nilai *compliance value*. Setelah didapat hasil normalisasi dari *compliance value*, langkah selanjutnya adalah mendapatkan rata-rata tiap level pada tiap sub domain untuk mendapatkan total rata-rata dalam sub domain tersebut, nilai rata-rata inilah yang dihitung sebagai *Total Maturity Level* [10].

Table 6. Total Maturity Level ME1 - Monitor and Evaluate IT Performance

ME1 LEVEL	NORMALIZED COMPLIANCE VALUE (b)					Maturity Level = b*Level					AVERAGE	TOTAL MATURITY LEVEL
	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5		
0	0.150	0.175	0.251	0.225	0.200	0.000	0.000	0.000	0.000	0.000	0.000	2.948
1	0.159	0.100	0.302	0.261	0.179	0.159	0.100	0.302	0.261	0.179	0.200	
2	0.179	0.128	0.204	0.310	0.179	0.358	0.256	0.409	0.620	0.358	0.400	
3	0.166	0.115	0.257	0.283	0.179	0.498	0.345	0.771	0.850	0.536	0.600	
4	0.190	0.102	0.249	0.249	0.175	0.761	0.409	0.998	0.998	0.701	0.773	
5	0.233	0.169	0.148	0.234	0.190	1.166	0.846	0.740	1.169	0.951	0.975	

Pada *tabel 6* diketahui *domain* ini (ME1) mendapatkan skor 2,948 dibulatkan menjadi skor 3 yang berarti *defined process* yang berarti Manajemen berkomunikasi dan melembagakan proses pemantauan standar. *Program* pendidikan dan pelatihan untuk pemantauan sudah diterapkan. Basis pengetahuan formal informasi kinerja historis mulai dikembangkan. Penilaian masih dilakukan di proses TI individu tingkat proyek dan tidak terintegrasi di antara semua proses. Alat untuk memantau proses dan layanan TI sudah didefinisikan. Pengukuran kontribusi fungsi layanan informasi terhadap kinerja organisasi sudah didefinisikan, menggunakan kriteria keuangan dan operasional tradisional. Pengukuran kinerja khusus TI, pengukuran non-keuangan, pengukuran strategis, pengukuran kepuasan pelanggan dan tingkat layanan sudah didefinisikan. Kerangka kerja didefinisikan untuk pengukuran pertunjukan [8].

Table 7. Total Maturity Level ME2 - Monitor and Evaluate Internal Control

ME2 LEVEL	NORMALIZED COMPLIANCE VALUE (b)					Maturity Level = b*Level					AVERAGE	TOTAL MATURITY LEVEL
	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5		
0	0.214	0.166	0.214	0.239	0.166	0.000	0.000	0.000	0.000	0.000	0.000	3.000
1	0.205	0.205	0.154	0.258	0.179	0.205	0.205	0.154	0.258	0.179	0.200	
2	0.176	0.157	0.176	0.295	0.196	0.352	0.313	0.352	0.591	0.391	0.400	
3	0.161	0.162	0.230	0.285	0.161	0.484	0.487	0.689	0.856	0.484	0.600	
4	0.175	0.189	0.203	0.258	0.175	0.700	0.756	0.811	1.032	0.700	0.800	
5	0.148	0.235	0.191	0.235	0.191	0.741	1.174	0.956	1.174	0.956	1.000	

Pada *tabel 7* diketahui *domain* ini (ME2) mendapatkan skor 3.00 yang berarti *defined process* adalah Manajemen mendukung dan melembagakan pemantauan pengendalian *internal*. Kebijakan dan prosedur dikembangkan untuk penilaian dan pelaporan atas aktivitas pemantauan pengendalian *internal*. Program pendidikan dan pelatihan untuk pemantauan pengendalian *internal* sudah ditetapkan. Sebuah proses didefinisikan untuk penilaian mandiri dan tinjauan jaminan kontrol *internal*, dengan peran untuk bisnis yang bertanggung jawab dan manajer TI. Peralatan sedang digunakan tetapi belum tentu terintegrasi ke dalam semua proses. Kebijakan penilaian risiko proses TI sedang digunakan di dalamnya kerangka kontrol yang dikembangkan secara khusus untuk organisasi TI. Risiko spesifik proses dan kebijakan mitigasi perlu ditentukan. [8].

Table 8 Total Maturity Level ME3 - Ensure Compliance With External Requirements

ME3 LEVEL	NORMALIZED COMPLIANCE VALUE (b)					Maturity Level = b*Level					AVERAGE	TOTAL MATURITY LEVEL
	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5		
0	0.181	0.181	0.181	0.275	0.181	0.000	0.000	0.000	0.000	0.000	0.000	3.000
1	0.157	0.105	0.317	0.263	0.157	0.157	0.105	0.317	0.263	0.157	0.200	
2	0.160	0.214	0.214	0.233	0.178	0.320	0.429	0.429	0.466	0.357	0.400	
3	0.134	0.298	0.188	0.298	0.081	0.403	0.895	0.565	0.895	0.242	0.600	
4	0.183	0.237	0.264	0.184	0.131	0.734	0.950	1.058	0.735	0.524	0.800	
5	0.108	0.257	0.285	0.189	0.161	0.538	1.287	1.426	0.943	0.807	1.000	

Pada *tabel 8* diketahui *domain* ini (ME3) mendapatkan skor 3.00 yang berarti *defined process* adalah Kebijakan, rencana dan prosedur sudah dikembangkan, didokumentasikan dan dikomunikasikan untuk memastikan kepatuhan terhadap peraturan dan kontrak dan kewajiban hukum, tetapi beberapa mungkin tidak selalu diikuti, dan beberapa mungkin sudah ketinggalan zaman atau tidak praktis untuk diterapkan. Ada sedikit pemantauan dilakukan dan ada persyaratan kepatuhan yang belum ditangani. Pelatihan diberikan dalam hukum eksternal dan persyaratan peraturan yang memengaruhi organisasi dan proses kepatuhan yang ditentukan. Kontrak *proforma* standar dan proses hukum sudah ada untuk meminimalkan risiko yang terkait dengan tanggung jawab kontraktual. [8].

Table 9 Total Maturity Level ME4 - Provide IT Governance

ME4 LEVEL	NORMALIZED COMPLIANCE VALUE (b)					Maturity Level = b*Level					AVERAGE	TOTAL MATURITY LEVEL
	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5		
0	0.047	0.287	0.287	0.189	0.189	0.000	0.000	0.000	0.000	0.000	0.000	3.000
1	0.084	0.238	0.255	0.238	0.186	0.084	0.238	0.255	0.238	0.186	0.200	
2	0.111	0.174	0.222	0.303	0.190	0.221	0.349	0.444	0.606	0.380	0.400	
3	0.155	0.268	0.222	0.222	0.133	0.465	0.803	0.667	0.667	0.398	0.600	
4	0.216	0.201	0.209	0.216	0.157	0.865	0.806	0.835	0.865	0.630	0.800	
5	0.209	0.241	0.139	0.233	0.178	1.047	1.204	0.696	1.165	0.888	1.000	

Pada *tabel 9* diketahui *domain* ini (ME4) mendapatkan skor 3.00 yang berarti *defined process* adalah Pentingnya dan perlunya tata kelola TI dipahami oleh manajemen dan dikomunikasikan kepada organisasi. Serangkaian dasar indikator tata kelola TI dikembangkan di mana hubungan antara ukuran hasil dan indikator kinerja didefinisikan dan didokumentasikan. Prosedur dibakukan dan didokumentasikan. Manajemen mengkomunikasikan prosedur standar, dan pelatihan. Perangkat IT diidentifikasi untuk membantu mengawasi tata kelola TI. Dasbor didefinisikan sebagai bagian dari kartu skor bisnis TI yang seimbang. Namun, diserahkan kepada individu untuk mendapatkan pelatihan, mengikuti standar dan menerapkannya. Proses dapat dipantau, tetapi penyimpangan, meskipun sebagian besar ditindaklanjuti oleh inisiatif individu, tidak mungkin terdeteksi oleh manajemen. [8].

### 3.4. Hasil Analisa Perbandingan Antara Manual Dengan Sistem

Tabel 10. Hasil Analisa Perbandingan antara Manual dengan Sistem.

NO	PERBANDINGAN	MANUAL	SISTEM
<b>Kecepatan</b>		<b>(Per Detik)</b>	
1	Kecepatan Input dalam satu kwitansi	60	10
2	Kecepatan Input Rekap Tagihan dalam satu tagihan	15	1
3	Kecepatan Pembuatan Rekap Penghuni Menunggak	15	1
<b>Ketepatan</b>		<b>Kemungkinan</b>	
1	Kesalahan Input (nama, blok, nominal) Dalam 100 Kwitansi	2	0
<b>Kekurangan dan Kelebihan</b>			
1	Dalam Pengeditan Data	Lebih Mudah, Karena di Excel Langsung ganti	Harus Membuka Menu Edit Tagihan IPL
2	Mengecek Tagihan Warga	Harus Melihat Daftar list dalam excel, dan terkadang belum update, karena banyak.	Automatis terupdate karena langsung menghitung tagihan dari awal.
3	Laporan Pemasukan Bulanan	Harus Dihitung manual terlebih dahulu menggunakan rumus SUM dalam Excel	Automatis Terbentuk berdasarkan uang masuk per bulannya.

Berdasarkan Tabel 10 dapat disimpulkan bahwa, berdasarkan pengamatan dan observasi langsung lapangan pada sales admin, penggunaan sistem IPL sangat membantu dalam kinerja sales admin dalam melaksanakan tugasnya untuk melakukan penagihan, pembuatan kwitansi dan pencatatan penghuni yang melakukan penunggakan iuran.

#### 4. KESIMPULAN

Berdasarkan hasil observasi langsung di lapangan, aplikasi Iuran Pengelolaan Lingkungan ini dapat mempermudah dan mempercepat admin dalam melakukan penginputan data, cek tagihan IPL, pembuatan tagihan, serta kwitansi IPL dengan lebih cepat dan mudah sesuai dengan form inputan Excel. Aplikasi ini telah diuji menggunakan *blackbox testing* dan dinyatakan dapat diterima dan diimplementasikan. Dari 8 pengujian menu menggunakan *blackbox testing*, dinyatakan 100% dapat diterima dan berfungsi dengan baik.

Hasil evaluasi SCRUM menunjukkan bahwa proyek ini selesai tepat waktu dengan total 3 sprint yaitu 6 minggu. Berdasarkan audit yang telah dilakukan, aplikasi ini berada pada level 3 (*defined*) dalam *maturity model framework* COBIT 4.1 domain monitor dan evaluasi kinerja TI. Pada level ini, prosedur penggunaan perangkat IT telah distandarisasi, didokumentasikan, dan dikomunikasikan melalui pelatihan. Prosedur ini harus diikuti oleh karyawan meskipun masih ada beberapa kekurangan dalam praktik IT formalitas.

Penilaian COBIT 4.1 menunjukkan bahwa Perumahan Metro Residence Bogor berada pada level 3 karena dari 5 responden, hanya 2 orang IT yang lebih memahami tentang tata kelola IT. Saran yang dapat diberikan untuk pengembangan ke depan adalah memperhalus tampilan perangkat lunak dengan menggunakan *template* terkini baik di aplikasi *mobile* maupun *web*-nya. *Website* admin dapat diganti dengan *framework* terbaru seperti React, Angular, atau Vue.js untuk performa yang lebih baik. *Backend framework* dapat diganti dengan bahasa pemrograman yang lebih *update* seperti Go atau Node.js agar data dan keamanan lebih *secure*. Aplikasi ini juga dapat dikembangkan agar penghuni dapat *login* dan melakukan pembayaran serta pengecekan mandiri pada aplikasi *mobile*.

Perlunya dilakukan audit periodik pada tim sales admin Perumahan Metro Residence untuk melihat peningkatan yang ada pada sistem dan sumber daya manusia. Penelitian ini dapat dikembangkan lebih lanjut dengan memasukkan domain *Plan and Organise* dari COBIT 4.1 untuk meningkatkan manajemen SDM. Pelatihan IT dan penambahan pengetahuan tentang tata kelola menjadi kunci utama untuk peningkatan skoring dari tiga menjadi lebih baik ke depannya.

## 5. DAFTAR PUSTAKA

- [1] E. Nur Maulana, D. Rahman Wijaya, K. Kunci, and P. Warga, “Aplikasi Warga Berseri Modul Pendataan Warga Dan Dashboard Monitoring Demografi Kependudukan Studi Kasus : Perumahan Permata Buah Batu,” vol. 7, no. 6, pp. 2684–2701, 2021.
- [2] E. Agustina and B. Dongoran, “The Effect of Application Technology Advances in Environmental Contribution Management at Gandaria Icon City Apartments (Pengaruh Kemajuan Teknologi Aplikasi Manyoption Dalam Pengelolaan Iuran Lingkungan Di Apartemen Gandaria Icon City),” vol. 6, no. 2, pp. 273–279, 2022, doi: 10.52362/jisamar.v6i2.724.
- [3] T. Tohirin, S. F. Utami, S. R. Widiyanto, and W. Al Mauludyansah, “Implementasi DevOps Pada Pengembangan Aplikasi e-Skrining Covid-19,” *Multinetics*, vol. 6, no. 1, pp. 15–20, 2020, doi: 10.32722/multinetics.v6i1.2764.
- [4] K. Schwaber And J. Sutherland, “The Scrum Guide The Definitive Guide To Scrum: The Rules Of The Game,” 2013.
- [5] Y. Christian and A. M. Pratama, “Perancangan Aplikasi Mindfulness Pada Platform Ios Untuk Pekerja Jarak Jauh Menggunakan Metodologi Agile Scrum,” *Conf. Manag. Business, Innov. Educ. Soc. Sci.*, vol. 1, no. 1, pp. 548–556, 2021.
- [6] H. R. Mustaqbal, M., Firdaus, R., & Rahmadi, “Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus Aplikasi Prediksi Kelulusan SNMPTN),” 2015.
- [7] K. Andry, J. F., & Christianto, “Audit Menggunakan COBIT 4.1 dan COBIT 5 Dengan Case Study. In Yogyakarta: TEKNOSAIN.,” 2018.
- [8] ISACA, “‘CoBIT 4.1,’ IT Gov. Inst., pp. 1–29,” 2007.
- [9] M. Yumna Majdina, I. B. Praptono, and M. Dellarosawati, “Perancangan Aplikasi Manajemen Persediaan Gudang Berbasis Website Pada Umkm Batik Sinuwun Dengan Agile Scrum Development Method Design of Web-Based Warehouse Management Application in Sinuwun Batik Sme Using Agile Scrum Development Method,” *Agustus*, vol. 7, no. 2, p. 5630, 2020.
- [10] J. M. Hudin, N. Anggraeny, and D. Prayudi, “Audit Sistem Informasi Inventory Pada PT. DP Utan Kayu Dengan Framework Cobit 4.1,” *J. Repos.*, vol. 2, no. 10, pp. 1359–1367, 2020, doi: 10.22219/repositor.v2i10.1015.