

# IMPLEMENTASI METODE SCRUM UNTUK MENDETEKSI KERUSAKAN GIGI BERBASIS MOBILE

Filfimo Yulfiz Ahsanul Hulqi<sup>1)</sup>, Ummi Athiyah\*<sup>2)</sup>, Novian Adi Prasetyo<sup>3)</sup>

1. Institut Teknologi Telkom Purwokerto, Indonesia
2. Institut Teknologi Telkom Purwokerto, Indonesia
3. Institut Teknologi Telkom Purwokerto, Indonesia

## Article Info

**Kata Kunci:** Android; Gigi; Kesehatan; Scrum.

**Keywords:** Android; Teeth; Healthcare; Scrum.

## Article history:

Received 20 November 2023

Revised 4 December 2023

Accepted 18 December 2023

Available online 1 March 2024

## DOI :

<https://doi.org/10.29100/jipi.v9i1.4399>

\* Corresponding author.

Ummi Athiyah

E-mail address:

[ummi@ittelmkom-pwt.ac.id](mailto:ummi@ittelmkom-pwt.ac.id)

## ABSTRAK

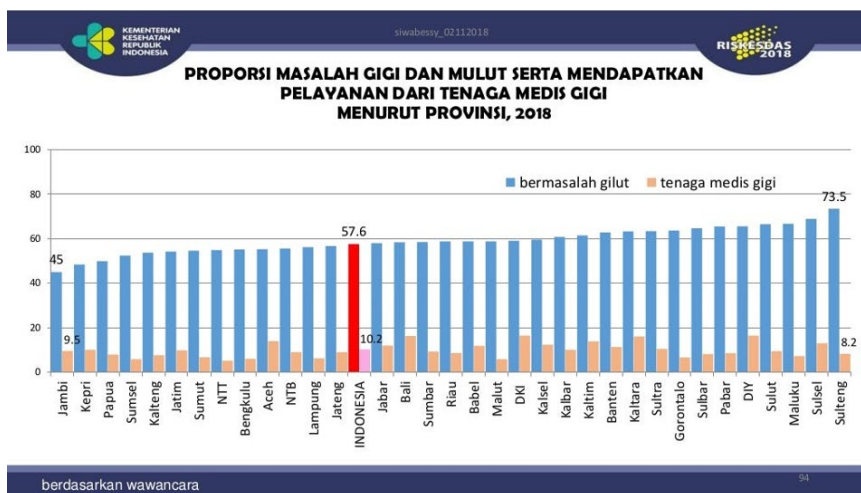
Kesehatan merupakan salah satu aspek yang sangat penting dalam kehidupan manusia, salah satunya adalah kesehatan gigi. Akses pemeriksaan untuk pencegahan dan pengobatan gigi masih kurang dan terbatas di berbagai wilayah, hal ini terbukti dari prevalensi penyakit gigi dan mulut di Indonesia terus bertambah. Riset Kesehatan Dasar pada tahun 2018 melakukan survei dengan sampel sebanyak 300.000 rumah tangga, mencatat bahwa proporsi terbesar masalah kesehatan di Indonesia adalah penyakit gigi dan mulut dengan proporsi masalah gigi dan mulut sebesar 57,6 % dan yang mendapat pelayanan kesehatan hanya sebesar 10,2%. Dari permasalahan tersebut dibuatlah sebuah aplikasi berbasis Android menggunakan metode Scrum untuk mendeteksi penyakit gigi dengan memanfaatkan kamera *smartphone*, yang dapat dilakukan secara mandiri sehingga dapat membantu masyarakat untuk mendiagnosis gejala awal penyakit gigi, sebelum berkonsultasi dengan dokter gigi. Hasil dari penelitian menunjukkan bahwa metode *Scrum* dapat diimplementasikan pada aplikasi, tampilan aplikasi dapat berkomunikasi dengan baik satu sama lain, dan fitur-fitur aplikasi dapat berjalan dengan baik berdasarkan hasil pengujian *UI* dan *Black Box Testing*.

## ABSTRACT

*Health is a very important aspect in human life, one of them is dental health. Access of examination for prevention and dental treatment still lacking and limited in various regions, it is proven from the prevalence of oral disease continues to grow. Riset Kesehatan Dasar in 2018 conduct a survey with a sample of 300.000 households, noting that the largest proportion of health problems in Indonesia is a oral disease with the proportion of dental and oral problems by 57,6% and only 10% receiving health services. From this problem an Android-based application will created using Scrum method to help detect dental disease by using smartphone camera, which can be done personally so that can help public to diagnoses early symptoms of dental disease before consulting with dentist. The result of this research indicate that the Scrum method can be implemented in the application, the application's interface can communicate effectively with each other, and the application features performing well, based on the result of UI and Black Box Testing.*

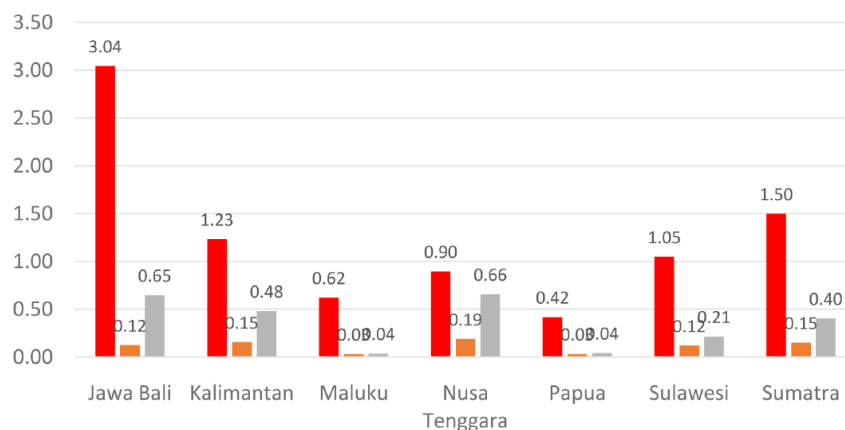
## I. PENDAHULUAN

**P**REVALENSI penyakit gigi dan mulut di Indonesia terus bertambah dikarenakan berbagai aspek salah satunya disebabkan oleh akses pemeriksaan untuk pencegahan dan pengobatan masih kurang dan terbatas di berbagai wilayah. Data statistik *Global Burden of Disease* mencatat bahwa penyakit gigi dan mulut merupakan penyakit konsukuen secara global yang dialami hampir seluruh penduduk dunia sebanyak 3,8 Milyar, terutama di Indonesia dengan sampel sebanyak 300.000 rumah tangga, yang dilakukan Riskesdas pada tahun 2018 mencatat bahwa proporsi terbesar masalah kesehatan gigi dan mulut sebesar 57,6 % dan yang mendapat pelayanan kesehatan hanya sebesar 10,2 %. Proporsi penduduk mempunyai masalah gigi dan mulut yang menerima pelayanan dari tenaga medis terdapat 13 provinsi di bawah rerata dan 21 provinsi di atas rerata seperti pada gambar 1.1 [1][2].



Gambar. 1. Proporsi masalah gigi dan mulut serta mendapatkan pelayan dari tenaga medis di Indonesia pada tahun 2018 [1]

Data dari Riskesdas tahun 2018 menunjukkan juga, bahwa penyebaran praktisi gigi di Indonesia kurang merata, yang dapat dilihat pada tabel gambar 2.



Gambar. 2. Penyebaran dokter gigi (merah), terapis gigi (oren), dan teknisi gigi (abu) [2]

Berdasarkan data tersebut, Indonesia membutuhkan sebuah aplikasi pelayanan kesehatan di bidang gigi dan mulut dengan membangun *Artificial Oral Screening* yaitu *Teledentistry* tanpa batasan jarak dengan menggunakan pengimplementasian *machine learning* dan kamera *smartphone* sebagai sistem *early sceening* yang dapat menjadi penghubung dokter gigi dan masyarakat. Aplikasi yang akan dibangun dapat menjadi solusi untuk dokter gigi yang terbatas dan kurang di Indonesia serta memastikan setiap penduduk di Indonesia memiliki akses yang adil terhadap pelayanan kesehatan, dan diharapkan dapat meningkatkan aksesibilitas masyarakat terhadap pelayanan kesehatan dan mencapai tujuan *suistanable development goals*, yaitu *good-health* dan *well-being* pada penanganan penyakit tidak menular.

Pada penelitian terdahulu dengan judul “Sistem Pakar Berbasis Android Untuk Diagnosa Penyakit Gigi Dan Mulut”. Pada penelitian ini mengimplementasikan sistem pakar berbasis Android, dengan hasil akurasi tingkat penerimaan *User Acceptance Test (UAT)* sebesar 93,03% [3].

Berdasarkan penjelasan yang telah dijelaskan di atas, pada pengerjaan penelitian ini, metode untuk mendeteksi kerusakan gigi akan menggunakan kamera *smartphone*. Kamera *smartphone* dipilih dalam pengerjaan penelitian ini karena pada saat ini *smartphone* sudah dapat dijangkau masyarakat [4]. Selain itu juga kamera dipilih untuk lebih memudahkan pengguna saat ingin melakukan pemeriksaan gigi secara mandiri dengan memanfaatkan kamera. Aplikasi yang akan dibangun ini juga memanfaatkan *TFLite* sebagai data dari model *machine learning* penyakit gigi yang akan dicocokkan dengan hasil gambar yang diambil oleh pengguna, setelah itu aplikasi akan menampilkan hasil dari deteksi gigi, nama penyakit gigi, deskripsi singkat mengenai penyakit gigi, dan saran pengobatan mandiri yang dapat dilakukan.

Berdasarkan latar belakang diatas, dapat diketahui permasalahan, bahwa prevalensi masalah kesehatan gigi masih mengalami pertumbuhan dikarenakan akses untuk pencegahan dan pengobatan gigi masih kurang dan terbatas di

beberapa wilayah di Indonesia.

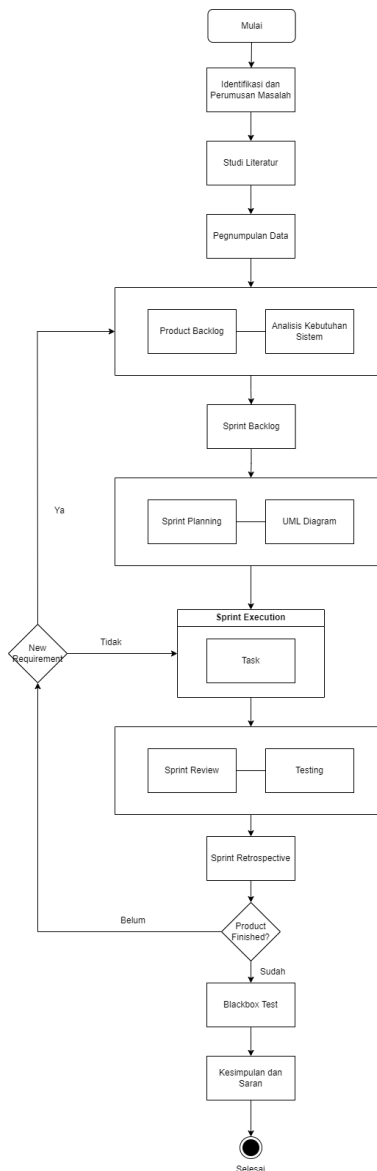
Berdasarkan uraian diatas maka diperoleh beberapa pertanyaan terkait, penelitian ini akan membahas dua pertanyaan terkait. Pertama, bagaimana membuat aplikasi Android untuk mendeteksi kerusakan gigi dengan mengimplementasikan metode Scrum dalam proses pengembangannya. Kedua. Bagaimana melakukan pengujian sistem pada aplikasi menggunakan *UI Test* dan *Black Box Testing*.

Penelitian ini diharapkan dapat membantu para masyarakat untuk mendiagnosis gejala awal penyakit gigi, sebelum berkonsultasi dengan dokter gigi. Penelitian ini akan mengimplementasikan model *machine learning* untuk mendeteksi kerusakan gigi berbasis Android, dengan memanfaatkan *library CameraX* dan *library TFLite*, sedangkan pada penelitian terdahulu berupa sistem pakar dan belum mengimplementasikan model *TFLite* sebagai model dari penyakit gigi.

## II. METODE PENELITIAN

### A. Diagram Alir Penelitian

Penelitian dimulai dengan identifikasi dan perumusan masalah, tujuan dari, tahapan selanjutnya adalah studi literatur dengan mempelajari literatur dari beberapa bidang ilmu yang berhubungan dengan penelitian seperti penggunaan *library* dan metode *Scrum*, tahapan selanjutnya adalah pengumpulan data dengan melakukan studi literatur dan observasi, tahapan selanjutnya adalah implementasi metode *Scrum*, tahapan selanjutnya adalah pengujian aplikasi, dan tahapan terakhir adalah penulisan laporan.



Gambar. 3. Diagram Alir Penelitian

## B. Penyakit Gigi

Pada aplikasi penelitian ini hanya dapat mengidentifikasi 2 macam penyakit saja, dikarenakan keterbatasan peneliti dalam mencari *dataset* penyakit gigi. Berikut adalah penyakit-penyakit yang dapat diidentifikasi :

### a. Periodontitis

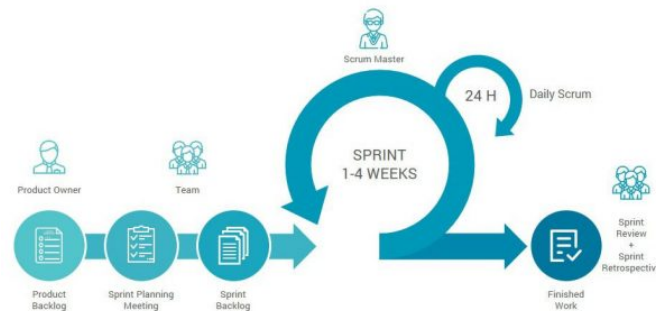
*Periodontitis* didefinisikan sebagai penyakit inflamasi dari jaringan pendukung gigi yang disebabkan oleh sebuah mikroorganisme spesifik atau sekumpulan mikroorganisme spesifik, sehingga menyebabkan terjadinya sebuah kerusakan progresif pada ligamen *periodontal* dan tulang alveolar dengan adanya pembentukan saku, resesi, atau bahkan keduanya [5]

### b. Teeth Discoloration

*Teeth Discoloration* disebabkan karena adanya *stain* (noda) yang menempel pada permukaan gigi. *Stain* merupakan masalah gigi estetik dan tidak menyebabkan peradangan pada gigitnya [6]. *Stain* dapat diklasifikasikan berdasarkan letaknya, yaitu *Intrinsic Discoloration* yang disebabkan oleh *chromogen* yang tersimpan ke sebagian besar gigi, dan *Extrinsic Discoloration* yang terletak pada permukaan gigi [7]. Selain itu *stain* juga dapat disebabkan oleh faktor lain seperti faktor usia (*aged-related*) [8].

## C. SCRUM

*Scrum* adalah sebuah proses untuk mengatur dan mengontrol pengembangan produk dan *software* [9]. *Scrum* merupakan perkembangan dari metodologi *Agile*, karena dasarnya mengimplementasikan filosofi *Agile*. Cara kerja dari *Scrum* adalah dengan membentuk sebuah tim untuk memberikan hasil kerja yang produktif, kreatif, memiliki nilai yang setinggi mungkin untuk dapat menyelesaikan masalah adaptif yang rumit [10].



Gambar. 4. Diagram Alir Penelitian [11]

Sebelum memulai *sprint*, tim akan menentukan *Product Backlog*, yang merupakan sebuah daftar dari seluruh kebutuhan yang harus dibangun dan diselesaikan. *Product Backlog* ini nantinya akan dibahas oleh tim pada tahap awal dari pengerjaan *sprint* [9], seperti yang tertera pada gambar 3.

### a. Sprint Planning

Tahapan awal dari *Scrum* dimana seluruh tim melakukan rapat perencanaan yang membahas hal yang perlu dilakukan pada *sprint* dan menyiapkan *sprint* untuk kedepannya [9].

### b. Daily Stand Up

Tahapan yang menjadi inti dari *Scrum*, dimana tim melakukan rapat setiap hari yang membahas mengenai kemajuan yang telah dilakukan dari masing-masing anggota tim [9].

### c. Sprint Demo

Pada tahapan ini masing-masing anggota tim menunjukkan hasil dari *blacklog* yang telah diselesaikan pada saat pengerjaan *sprint* [9].

### d. Sprint Retrospective

Setelah keseluruhan *sprint* selesai, diakhiri dengan melakukan retrospektif. Masing-masing anggota tim akan mengevaluasi *sprint* yang telah selesai dilakukan sebelumnya, dan memberikan *feedback* untuk mendapatkan kinerja yang lebih baik [9].

Dalam *Scrum* memiliki tiga pilar yang harus diterapkan untuk memperkokoh setiap implementasi dari proses empiris adalah [12]:

### a. Transparansi

Informasi yang relevan tentang tugas dan proses harus tersedia dan dapat diakses oleh Tim *Scrum*. Hal ini mencakup pemahaman yang jelas tentang tujuan, tugas yang harus dikerjakan, serta kemajuan dan kendala yang dihadapi [12].

b. Inspeksi

Tim secara teratur melakukan evaluasi terhadap pekerjaan yang dilakukan untuk mengidentifikasi permasalahan, kesalahan, peluang perbaikan. Tujuan dari inspeksi untuk meningkatkan kualitas dan efektivitas kerja [12].

c. Adaptasi

Tim mengambil tindakan perbaikan berdasarkan hasil inspeksi. Jika ada masalah atau perubahan, tim melakukan penyesuaian dan perubahan diperlukan untuk mencapai hasil yang diinginkan [12].

D. UI Test

*UI test* digunakan untuk menguji kesesuaian aplikasi dengan kondisi pengguna ketika berinteraksi pada sebuah aplikasi untuk memastikan apakah fungsi dari tampilan aplikasi sudah sesuai. Pengujian ini dapat dilakukan dengan dua cara, dengan manual atau secara otomatis dengan memanfaatkan beberapa *frameworks* [13].

E. Black Box Testing

*Black box testing* digunakan untuk mengamati hasil dari *input* dan *output* dari perangkat lunak yang dibuat. Pengujian ini dilakukan pada akhir pembuatan perangkat lunak untuk mengetahui fungsi dari perangkat lunak tersebut sudah baik atau belum [14].

### III. HASIL DAN PEMBAHASAN

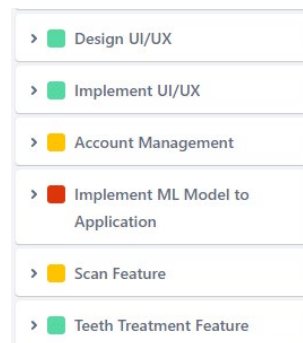
Pada bab ini membahas keseluruhan proses dari pembuatan, pengujian, dan hasil dari aplikasi yang sudah dibuat menggunakan metode *Scrum*. Sebelum memulai pembuatan aplikasi, penulis wajib mendefinisikan seluruh kebutuhan yang dibutuhkan, sehingga dengan ini akan memperjelas apa saja yang harus dilakukan sebelum pengembangan dimulai.

#### A. SCRUM

a. Product Backlog

1. Pendefinisian Epic

Pendefinisian *epic* didasarkan pada kebutuhan aplikasi, seperti, *ui/ux*, autentifikasi, peoses pemeriksaan gigi, dan penjabaran saran yang diterima. *Epic* dijabarkan lebih lanjut pada gambar 4, dibawah ini.



Gambar. 5. Diagram Alir Penelitian

2. Identifikasi Issue

Setelah dilakukan *brainstorming*, didapatkan *issue backlog*, yang sudah dikelompokkan berdasarkan *epic*, dapat dilihat pada gambar 5.



TA-20	Menampilkan hasil Treatment	TREATMENT FEATURE	1	TO DO	100%
TA-21	Menyimpan hasil Scan	ACCOUNT MANAGEMENT	2	TO DO	100%
TA-22	Menampilkan hasil Scan	ACCOUNT MANAGEMENT	2	TO DO	100%
TA-23	Menampilkan detail Scan History	ACCOUNT MANAGEMENT	2	TO DO	100%
TA-24	Menampilkan Home	ACCOUNT MANAGEMENT	1	TO DO	100%
TA-25	Testing		3	TO DO	100%

Gambar. 6. Identifikasi dan Pemberian *Point* pada *Issue Backlog*

Setelah didapatkan keseluruhan *issue backlog*, beri nilai untuk masing-masing *issue backlog*. Nilai yang semakin tinggi mengartikan bahwa *issue* tersebut semakin lama untuk dikerjakan atau semakin sulit untuk diselesaikan. Hasil dari pemberian nilai *issue* dapat dilihat pada gambar 5.

Nilai *point* yang diberikan diatas, didasarkan dari pengalaman pribadi penulis dalam mengembangkan aplikasi Android sebelumnya, nilai *point* yang rendah menggambarkan bahwa pengerjaan *task* tersebut dapat dengan cepat untuk diselesaikan, sebaliknya nilai *point* yang tinggi menggambarkan bahwa pengerjaan *task* tersebut memakan waktu yang cukup lama untuk diselesaikan. Berikut adalah ketentuan dalam pemberian nilai untuk setiap *task*.

- Nilai 1 diberikan untuk *task* yang sekiranya dapat diselesaikan dalam waktu 1 hari.
- Nilai 2 diberikan untuk *task* yang sekiranya dapat memakan waktu 2-3 hari.
- Nilai 3 diberikan untuk *task* yang membutuhkan waktu cukup lama, dikarenakan proses pengimplementasiannya belum pernah dilakukan atau perlu melakukan pembelajaran terlebih dahulu.

## b. Sprint Backlog

### 1. Penamaan *Sprint*

Tidak ada aturan penamaan khusus dalam penamaan sebuah *sprint*, dalam penelitian ini untuk penamaan *sprint* penulis menggunakan urutan abjad. Penamaan *sprint* dapat dilihat pada gambar 6.

TA A	Add dates (0 issues)	Start sprint
TA B	Add dates (0 issues)	Start sprint
TA C	Add dates (0 issues)	Start sprint

Gambar. 7. Penamaan *Sprint*

### 2. Perencanaan Rentang *Sprint*

Rencana awal untuk setiap *sprint* akan memiliki rentang waktu pengerjaan selama satu minggu, dimulai dari tanggal 6 Maret untuk *sprint* pertama, dan berakhir pada 27 Maret pada *sprint* terakhir. Perencanaan rentang waktu dapat dilihat pada gambar 7.

TA A	6 Mar – 13 Mar (0 issues)	Start sprint
TA B	13 Mar – 20 Mar (0 issues)	Start sprint
TA C	20 Mar – 27 Mar (0 issues)	Start sprint

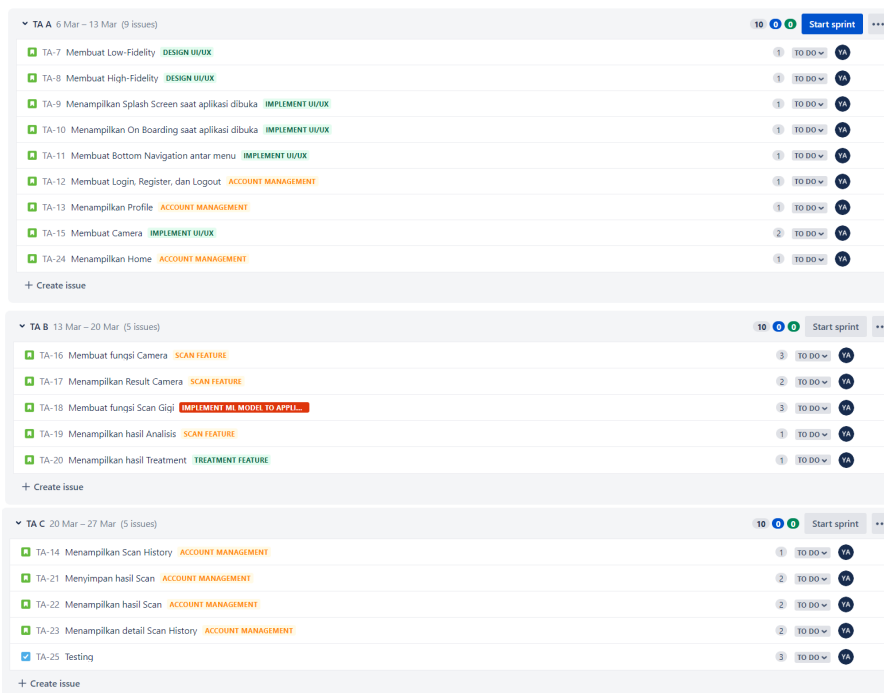
Gambar. 8. Rentang Waktu *Sprint*

### 3. Perencanaan Pembagian *Sprint*

Penentuan seberapa banyak *sprint* yang akan dilakukan saat proses pengembangan, dengan menentukan terlebih dahulu batas *point* untuk setiap *sprint*. Total *sprint* didapat dari rumus :

$$\text{“Jumlah sprint} = \frac{\text{total story point}}{\text{batas story point tiap sprint}} \text{” [15].}$$

Dalam penelitian ini, batas *story* untuk setiap *sprint* adalah 10 *point*. Dikarenakan total dari keseluruhan *story point* adalah 30, sehingga jumlah *sprint* yang akan dilakukan adalah sebanyak 3 kali. Masukkan tiap-tiap *issue* kedalam *sprint*, pehitungkan tingkat prioritas dari sebuah *issue*, dahulukan *issue* yang memiliki prioritas tinggi dalam pengerjaan, lakukan pembagian *issue* ini sampai semua *issue* yang telah didefinisikan sebelumnya ter-cover seluruhnya dalam masing-masing *sprint*. Berikut adalah hasil dari pengelompokkan *issue* kedalam *sprint*, dapat dilihat pada gambar 8.



Gambar. 9. Issue Backlog kedalam Sprint

### c. Sprint Execution

#### 1. Sprint A

Hasil eksekusi dari *sprint* A yang dimulai dari tanggal 06 Maret 2023 – 12 Maret 2023, dapat direpresentasikan dengan tabel I dan diagram grafik 1.

TABEL I  
SPRINT A

ID	Task	Story Est	06-Mar-	07-Mar-	08-Mar-	09-Mar-	10-Mar-	11-Mar-	12-Mar-
			23 Senin	23 Selasa	23 Rabu	23 Kamis	23 Jumat	23 Sabtu	23 Minggu
TA-7	Membuat Low-Fidelity	1	1						
TA-8	Membuat High-Fidelity	1	1						
TA-9	Menampilkan Splash Screen saat aplikasi dibuka	1		1					
TA-10	Menampilkan On Boarding saat aplikasi dibuka	1		1					
TA-11	Membuat Bottom Navigation antar menu	1		1					
TA-12	Membuat Login, Register, dan Logout	1			1				
TA-13	Menampilkan Profile	1			1				
TA-15	Membuat Camera	2			1	1			
TA-24	Menampilan Home	1			1				
	Remaining Effort	10	8	5	1	0	0	0	0
	Ideal Effort	10	9	7	6	4	3	1	0

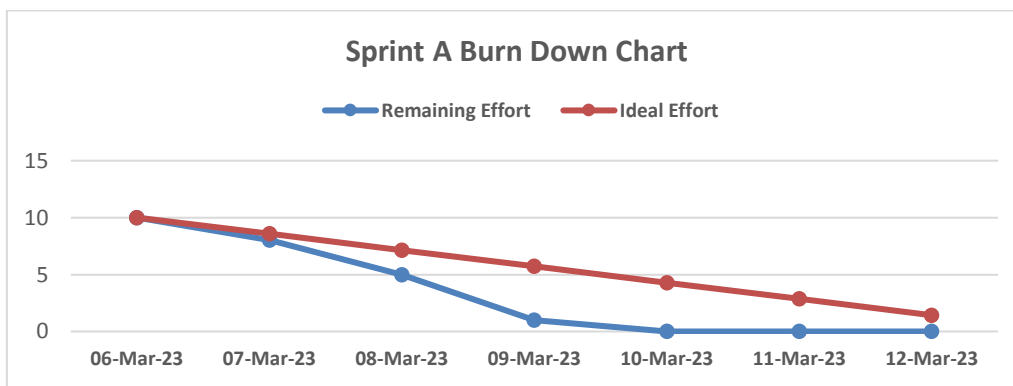


Diagram Grafik 1 Burndown Chart A

- **Sprint Review**  
 Berdasarkan pada tabel I dan diagram grafik 1, dari tanggal 06 Maret 2023 – 12 Maret 2023, pengerjaan *sprint* dapat dikatakan cepat, semua *task* dapat dikerjakan dengan sangat baik sebelum *deadline sprint* pada tanggal 12 Maret 2023. Pada tanggal 08 Maret 2023 terdapat pengerjaan *task* yang besar dimana dalam hari tersebut terdapat 3 *task* yang dapat diselesaikan, dan memiliki nilai 4 *point*. Untuk garis *remaining effort* berada dibawah *ideal effort* yang menandakan bahwa pengerjaan pada *sprint* tersebut berjalan cepat.
- **Sprint Retrospective**  
 Berdasarkan penjelasan pada *sprint review* dapat disimpulkan bahwa kinerja dari *sprint* tersebut dapat dikatakan sangat baik, terbukti dari garis *remaining effort* yang berada dibawah *ideal effort*. Dan *sprint* dapat diselesaikan lebih cepat, yaitu dapat diselesaikan pada tanggal 09 Maret 2023, lebih cepat 4 hari sebelum batas *deadline* yang telah ditentukan pada tanggal 12 Maret 2023. Diharapkan untuk kinerja seperti ini dapat dipertahankan untuk pengerjaan *sprint* berikutnya.

## 2. Sprint B

Mempertimbangkan hasil dari pengerjaan *sprint* A sebelumnya, yang dapat diselesaikan pada tanggal 09 Maret 2023, penulis melakukan penyesuaian pada *sprint* B, yaitu memperbarui tanggal mulai *sprint* B yang sebelumnya direncanakan pada tanggal 13 Maret 2023 – 20 Maret 2023 menjadi 11 Maret 2023 – 17 Maret 2023. Hasil eksekusi dari *sprint* B, dapat direpresentasikan dengan tabel *burndown chart* pada tabel II, dan diagram grafik 2.

TABEL II  
SPRINT B

ID	Task	Story Est	11-Mar-23	12-Mar-23	13-Mar-23	14-Mar-23	15-Mar-23	16-Mar-23	17-Mar-23
			Sabtu	Minggu	Senin	Selasa	Rabu	Kamis	Jumat
TA-16	Membuat fungsi Camera	3	1			2			
TA-17	Menampilkan Result Camera	2				1		1	
TA-18	Membuat fungsi Scan Gigi	3				1			2
TA-19	Membuat hasil Analisis	1						1	
TA-20	Membuat hasil Treatment	1						1	
Remaining Effort		10	9	9	9	5	5	2	0
Ideal Effort		10	9	7	6	4	3	1	0



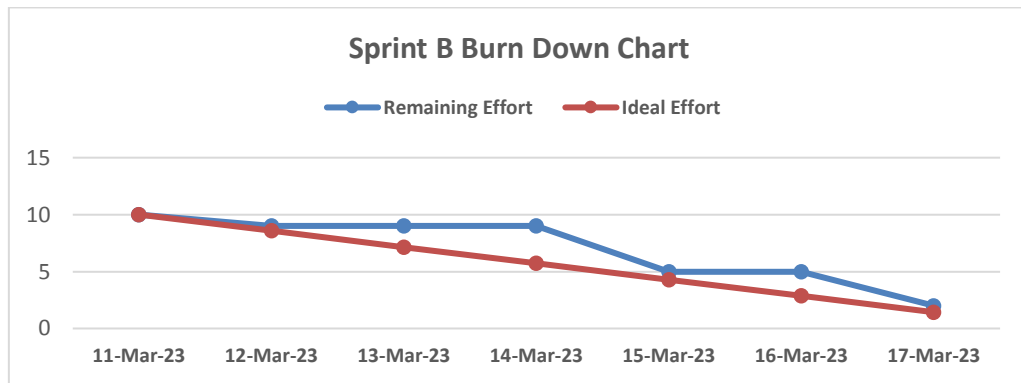


Diagram Grafik 2 Burndown Chart B

- Sprint Review

Berdasarkan pada tabel II, dan diagram grafik 2, dari tanggal 11 Maret 2023 – 17 Maret 2023, pengerjaan *sprint* dapat dikerjakan dengan baik, semua *task* dapat dikerjakan dengan cukup baik, walaupun sempat mengalami kendala saat pengerjaan, terbukti pada *burndown chart*, dimana pada tanggal 12 Maret 2023 garis dari *remaining effort* mulai berada diatas *ideal effort*.

Pada proses pengerjaan TA-16, penulis memerlukan waktu untuk mempelajari terlebih dahulu untuk membuat *camera* pada Android, dikarenakan menggunakan *library CameraX*. Penggunaan dari *library* ini cocok digunakan dengan kebutuhan aplikasi dimana membutuhkan kustomisasi penuh terhadap kamera Android. Untuk pengerjaan TA-17 menjadi terhambat dikarenakan *task* tersebut berhubungan dengan TA-16, yaitu untuk menampilkan hasil dari pengambilan gambar.

Pada proses pengerjaan TA-18, penulis kembali lagi memerlukan waktu untuk mempelajari terlebih dahulu untuk membuat sebuah fungsi untuk melakukan *scan* gigi dari hasil pengambilan gambar dengan memanfaatkan *library TFLite*, dengan menggunakan *TFLite*, model dari gambar gigi yang telah di-*training* dapat dijalankan didalam aplikasi secara langsung, dengan begitu proses *scan* gigi akan lebih cepat.

Untuk TA-19 dan TA-20, dapat diselesaikan lebih cepat, tetapi penulis memutuskan untuk mengerjakannya setelah TA-16 dan TA-17 selesai dikarenakan keseluruhan *task* tersebut saling berhubungan.

- Sprint Retrospective

Berdasarkan penjelasan pada *sprint review* dapat disimpulkan bahwa kinerja dari *sprint* tersebut dapat dikatakan cukup baik, karena *sprint* dapat diselesaikan tepat waktu sesuai dengan *deadline* yang ditentukan yaitu pada tanggal 17 Maret 2023. Pengerjaan *sprint* kali ini sering mengalami kendala terbukti dari garis *remaining effort* yang selalu berada diatas *ideal effort*. Dari *sprint A* sebelumnya dapat diketahui bahwa *sprint B* kali ini mengalami penurunan kinerja, diharapkan untuk *sprint* berikutnya kinerja pengerjaan dapat diperbaiki menjadi lebih baik lagi.

### 3. Sprint C

Dikarenakan pengerjaan *sprint A* yang cepat, dan penyesuaian pada *sprint B*. Maka *sprint C* juga melakukan penyesuaian yang serupa, yaitu memperbarui tanggal mulai *sprint C* yang sebelumnya direncanakan pada tanggal 20 Maret 2023 – 27 Maret 2023, menjadi 18 Maret 2023 – 23 Maret 2023. Hasil eksekusi dari *sprint C*, dapat direpresentasikan dengan tabel *burndown chart* seperti tabel III, dan diagram grafik 3.

TABEL III  
 SPRINT C

ID	Task	Story Est	18-Mar-23	19-Mar-23	20-Mar-23	21-Mar-23	22-Mar-23	23-Mar-23	24-Mar-23
			Sabtu	Minggu	Senin	Selasa	Rabu	Kamis	Jumat
TA-14	Menampilkan Scan History	2			1	1			
TA-21	Menampilkan hasil Scan	1	1						
TA-22	Menyimpan hasil Scan	2		2					
TA-23	Menampilkan detail Scan History	2				1	1		
TA-25	Testing	3						1	2
Remaining Effort		10	9	9	9	5	5	2	0
Ideal Effort		10	9	7	6	4	3	1	0

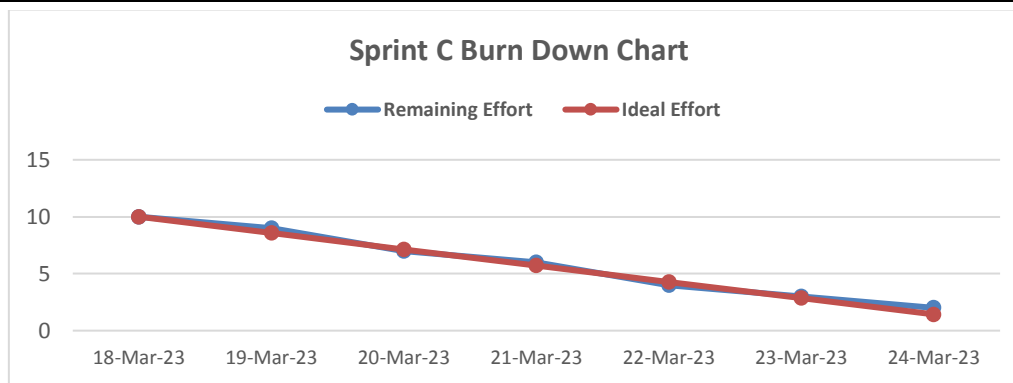


Diagram Grafik 3 Burndown Chart C

- Sprint Review

Berdasarkan pada tabel III dan diagram grafik 3, dari tanggal 18 Maret 2023 – 24 Maret 2023, pengerjaan *sprint* kali ini dapat dikatakan baik. Pada proses pengerjaan TA-21 dan TA-22, penulis terlalu meng-*overestimate task* tersebut, dimana penulis memiliki bayangan bahwa kedua *task* tersebut akan terkendala dan memakan waktu yang lama, kenyataannya *task* tersebut dapat diselesaikan dalam waktu masing-masing 1 hari.

Pada proses pengerjaan TA-25, penulis mengalami kendala yang cukup lama dikarenakan penulis membutuhkan waktu untuk melakukan *testing*, dimana responden utama memerlukan waktu untuk mengatur jadwal temu.

Untuk proses pengerjaan TA-14 dan TA-23, *task* tersebut dapat diselesaikan dengan normal sesuai dengan perkiraan penulis.

- Sprint Retrospective

Berdasarkan penjelasan pada *sprint review* dapat disimpulkan bahwa kinerja dari *sprint* tersebut dapat dikatakan stabil dan baik karena waktu pengerjaan *sprint* dapat diselesaikan seperti kedua *sprint* sebelumnya *sprint A*, dan *sprint B* yang dapat diselesaikan dalam waktu 7 hari.

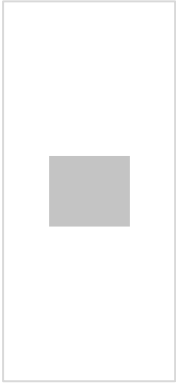
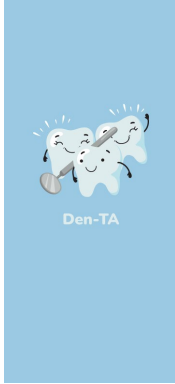

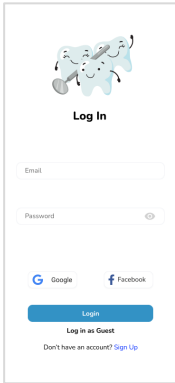
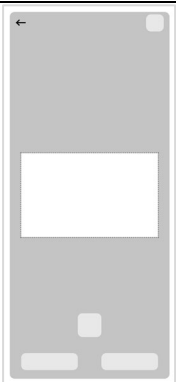
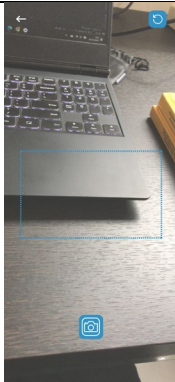


Dari keseluruhan *sprint* yang dilakukan, didapatkan sebuah hasil bahwa *sprint A* menunjukkan kinerja yang sangat baik dimana dapat diselesaikan lebih cepat dari estimasi pengerjaan, kemudian *sprint B* sempat menghadapi kendala dan kesulitan dalam mengerjakan *task* tetapi masih dapat mencapai hasil yang baik dimana dapat diselesaikan sesuai dengan estimasi pengerjaan, dan *sprint C* menunjukkan kinerja stabil dengan baik dimana dapat diselesaikan sesuai dengan estimasi pengerjaan.

d. Hasil Produk


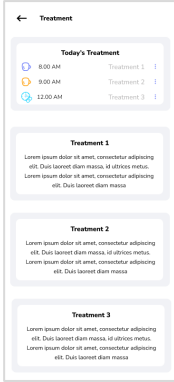
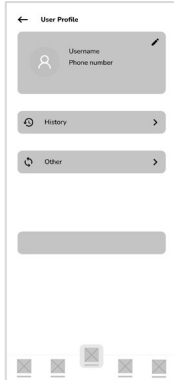
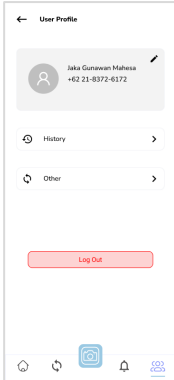

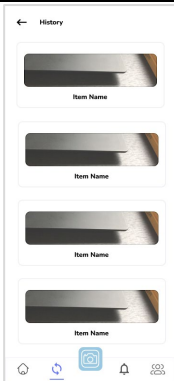
Berikut adalah hasil dari rancangan dan hasil dari aplikasi yang telah dibuat.

1. Rancangan Produk

TABEL IV  
 DESAIN LO-FI DAN HI-FI

No	Desain Lo-Fi	Desain Hi-Fi
1		
	Gambar. 10. <i>Lo-Fi</i> Splash Screen	Gambar. 11. <i>Hi-Fi</i> Splash Screen
2		
	Gambar. 12. <i>Lo-Fi</i> Log In	Gambar. 13. <i>Hi-Fi</i> Log In
3		
	Gambar. 14. <i>Lo-Fi</i> Camera	Gambar. 15. <i>Hi-Fi</i> Camera
4		
	Gambar. 16. <i>Lo-Fi</i> Home	Gambar. 17. <i>Lo-Fi</i> Home

TABEL V  
 DESAIN LO-FI DAN HI-FI (LANJUTAN)

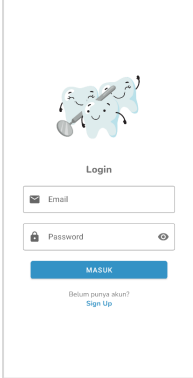
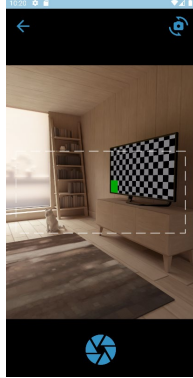


No	Desain Lo-Fi	Desain Hi-Fi
5		
	Gambar. 18. <i>Lo-Fi Self Care</i>	Gambar. 19. <i>Hi-Fi Self Care</i>
6		
	Gambar. 20. <i>Lo-Fi Profile</i>	Gambar. 21. <i>Hi-Fi Profile</i>
7		
	Gambar. 22. <i>Lo-Fi History</i>	Gambar. 23. <i>Lo-Fi History</i>

Pada tabel IV dan V diatas merupakan desain rancangan awal dari aplikasi yang akan dibuat dimana terbagi menjadi dua desain, yang pertama adalah desain *low fidelity (lo-fi)* yang merupakan sebuah representasi yang masih berupa desain sederhana dan kasar dari produk yang akan dibuat, yang kedua adalah *high fidelity (hi-fi)* yang merupakan desain lanjutan dari *low fidelity* yang sudah diperhalus dan lebih kompleks dengan menunjukkan pemilihan *font, icon*, warna dan ilustrasi yang akan digunakan.

## 2. Implementasi Produk

Berikut adalah hasil akhir dari aplikasi yang diimplementasikan yang telah mengalami perubahan berdasarkan kekurangan dari desain sebelumnya. Dapat dilihat pada tabel VI dan tabel VII.

TABEL VI  
 HASIL AKHIR IMPLEMENTASI PRODUK

No	Hasil Desain	Keterangan
1		<p>Gambar 24 disamping merupakan hasil akhir dari tampilan <i>log in</i>. Pada <i>log in</i> mengalami perubahan dimana pada hasil akhir untuk melakukan <i>log in</i> hanya menggunakan <i>authentication email</i> yang telah didaftarkan sebelumnya.</p>
2		<p>Gambar 25 disamping merupakan hasil akhir tampilan <i>camera</i>. Pada <i>camera</i> hanya mengalami perubahan <i>layout</i> dan <i>icon</i>.</p>
3		<p>Gambar 26 disamping merupakan hasil akhir dari tampilan <i>home</i> saat pengguna sudah melakukan <i>scan gigi</i>. Perubahan yang dilakukan hanya mengganti warna dari <i>border card</i> sesuai dengan hasil <i>self care</i>.</p>
4		<p>Gambar 27 disamping merupakan hasil akhir dari tampilan <i>self care</i>. Pada <i>self care</i> hanya mengalami perubahan dibagian <i>card border</i> yang disesuaikan dengan hasil analisis gigi.</p>

Gambar 24. Halaman *Log In*

Gambar 25. Halaman *Camera*

Gambar 26. Halaman *Home*

Gambar 27. Halaman *Self Care*

TABEL VII  
 HASIL AKHIR IMPLEMENTASI PRODUK (LANJUTAN)

No	Desain Lo-Fi	Desain Hi-Fi
5		<p>Gambar 28 disamping merupakan hasil akhir tampilan <i>profile</i>. Pada <i>profile</i> mengalami beberapa perubahan, pada bagian <i>card profile</i>, dan untuk <i>button</i> beberapa dihilangkan dan mengganti <i>style</i> dari <i>button</i>.</p>
7		<p>Gambar 29 disamping merupakan hasil akhir tampilan <i>history</i>. Pada <i>history</i> hanya mengalami perubahan pada <i>item title</i> yang sebelumnya berada pada bagian bawah.</p>

Gambar. 28. Halaman Profile

Gambar. 29. Lo-Fi History

## B. Testing

### a. UI Test

UI Test dilakukan terhadap tampilan yang telah dibuat pada aplikasi, diantaranya sebagai berikut yang dapat dilihat pada tabel VIII.

TABEL VIII  
 UI TESTING

No	Screen	MethodTest	MethodTest	Result
1	LoginActivity	LoginActivityTest	testMoveToSinUpActivity testLoginSession	Valid Valid
2	SignUpActivity	SignUpActivityTest	testMoveToLoginAcitivity testSignUpSession	Valid Valid
3	MainActivity	MainActivityTest	testMoveToResultCamera testMoveToProfile testMoveToHome	Valid Valid Valid
4	CameraResultActivity	CameraResultActivityTest	testMoveToCameraActivity	Valid
5	CameraActivity	CameraActivityTest	testButtonCapture testCheckButton testRetakeButton testChangeCameraButton	Valid Valid Valid Valid
6	ProfileFragment	ProfileFragmentTest	testLogoutButton testHistoryButton	Valid Valid

### b. Black Box Testing

Setelah proses *Scrum* selesai dilakukan, tahapan pengujian *black box testing* dilakukan untuk memverifikasi bahwa aplikasi yang telah dibuat memiliki fungsionalitas yang sesuai dan berjalan tanpa kesalahan, sesuai dengan kebutuhan sistem yang telah dirancang sebelumnya. Pengujian ini menggunakan kuesioner 25 Poin yang harus dijawab sebagai “*valid*” atau “*invalid*” pada kolom “hasil pengujian”.

Pengujian *black box* dilakukan terhadap 10 responden, yang terdiri dari 1 dokter gigi, 1 dosen, 2 masyarakat umum, dan 6 mahasiswa. Dari 25 Poin yang diujikan kepada 10 responden, seluruh poin mendapatkan jawaban “*valid*” pada kolom “hasil pengujian sesuai”. Dari hasil ini dapat disimpulkan bahwa aplikasi yang dibuat memiliki fungsionalitas yang sesuai dan berjalan dengan baik.

Berdasarkan keseluruhan penjelasan *sprint A*, *sprint B*, dan *sprint C*, dapat ditarik sebuah kesimpulan bahwa pengerjaan dari masing-masing *sprint* dapat berjalan dengan baik. Untuk pengerjaan aplikasi menggunakan metode *Scrum* dapat disimpulkan berjalan dengan lancar, dan metode *Scrum* cocok digunakan pada pengerjaan aplikasi ini, dimana sangat sesuai dengan kondisi saat pengerjaan aplikasi yang seringkali mengalami perubahan ditengah pengerjaan, sehingga memberikan fleksibilitas yang tinggi untuk memenuhi kebutuhan aplikasi bersamaan dengan pengerjaan yang terus berjalan.

#### IV. KESIMPULAN

Hasil penelitian yang dilakukan oleh penulis untuk pembuatan aplikasi Android dengan Implementasi Metode Scrum Untuk Mendeteksi Kerusakan Gigi Berbasis Mobile adalah sistem aplikasi android untuk mendeteksi kerusakan gigi dengan memanfaatkan *library CameraX* dan *TFLite* dapat diimplementasikan dengan baik dan berhasil dikembangkan menggunakan metode Scrum. Berdasarkan hasil dari *ui testing* yang dilakukan, keseluruhan tampilan dapat berkomunikasi satu sama lain dengan baik. Dan berdasarkan hasil dari *black box testing* yang dilakukan, keseluruhan fitur dapat berjalan dengan baik.

#### DAFTAR PUSTAKA

- [1] P. R. Sri Susilawati, drg, “Implementation - National Basic Health Research ( RISKESDAS ) 2018,” *Handbook*, 2018.
- [2] N. R. P. Gofur, A. Z. Z. Aghasy, and A. R. P. Gofur, “Spatial distribution analysis of dentists, dental technicians, and dental therapists in Indonesia,” *F1000Research*, vol. 10, 2021, doi: 10.12688/f1000research.50869.1.
- [3] H. S. Arfajsyah, I. Permana, and F. N. Salisah, “Sistem Pakar Berbasis Android Untuk Diagnosa Penyakit Gigi Dan Mulut,” *J. Ilm. Rekayasa dan Manaj. Sist. Inf.*, vol. 4, no. 2, p. 110, 2018, doi: 10.24014/rmsi.v4i2.5678.
- [4] E. I. Setiawan, H. K. B. Prakoso, T. P. Gunawan, E. Setyati, and J. Santoso, “Aplikasi Mobile Untuk Memantau Body Mass Index Dengan Metodologi Scrum,” *Teknika*, vol. 10, no. 3, pp. 242–250, 2021, doi: 10.34148/teknika.v10i3.405.
- [5] R. Suresh, *Aging and periodontal disease*. 2006.
- [6] A. Juniafti, “Pengaruh Merokok dan Minuman Berwarna Terhadap pembentukan stain,” *J. Inf.*, vol. 10, no. 1969, pp. 1–16, 2013, [Online]. Available: oai:repository.unhas.ac.id:123456789/7963.
- [7] S. Manuel, P. Abishek, and M. Kundabala, “Etiology of tooth discoloration- a review,” *Niger. Dent. J.*, vol. 18, no. 2, pp. 56–63, 2010.
- [8] “Tooth Discoloration: Causes, Prevention, How to Remove Stains.” <https://www.healthline.com/health/tooth-discoloration#discoloration-types> (accessed Oct. 23, 2022).
- [9] J. Sutherland, *More Praise for Scrum : The Art of Doing Twice the Work in Half the Time*. 2014.
- [10] “Berkenalan dengan Scrum. Apa itu Scrum? | by Panca Sitorus | Easyread.” <https://medium.easyread.co/framework-scrum-d3a11ae1629d> (accessed Oct. 23, 2022).
- [11] “Metode Scrum: Arti, Cara Kerja, Peran-Peran, dan Manfaat - Glints Blog.” <https://glints.com/id/lowongan/metode-scrum/> (accessed Jun. 27, 2023).
- [12] E. al. Sibarani, “Panduan Scrum,” *Imp. J. Interdiscip. Res.*, vol. 2, no. 12, pp. 293–298, 2017, [Online]. Available: <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Indonesian.pdf>.
- [13] R. Thakkar, “UI Testing with Espresso in Android | by Rina Thakkar | Mindful Engineering | Medium.” <https://medium.com/mindful-engineering/ui-testing-with-espresso-in-android-10dfbc9f25da> (accessed May 22, 2023).
- [14] S. Nidhra, “Black Box and White Box Testing Techniques - A Literature Review,” *Int. J. Embed. Syst. Appl.*, vol. 2, no. 2, pp. 29–50, 2012, doi: 10.5121/ijesa.2012.2204.
- [15] D. Pangestu, “Pengembangan Fitur Push Notification Pada Aplikasi IGracias Mobile Menggunakan Metode Scrum,” 2021.