



**RANCANG BANGUN APLIKASI *SELF PRODUCTIVITY LIST*
DALAM MENGATUR AKTIVITAS HARIAN BERBASIS
ANDROID**

Disusun Sebagai Salah Satu Syarat Untuk Menempuh Ujian Akhir
Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi Medan

SKRIPSI

OLEH

NAMA : LIANA
N. P. M : 1614370190
PROGRAM STUDI : SISTEM KOMPUTER

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2021**

RANCANG BANGUN APLIKASI *SELF PRODUCTIVITY LIST* DALAM MENGATUR AKTIVITAS HARIAN BERBASIS ANDROID

Disusun Sebagai Salah Satu Syarat Untuk Menempuh Ujian Akhir
Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi Medan

SKRIPSI

OLEH

NAMA : LIANA
N. P. M : 1614370190
PROGRAM STUDI : SISTEM KOMPUTER

**Skripsi telah disetujui oleh Dosen Pembimbing Skripsi
pada tanggal 18 Februari 2021 :**

Dosen Pembimbing I



(Hafni, S.Kom., M.Kom)

Dosen Pembimbing II



(Hermansyah, S.Kom., M.Kom)

Mengetahui,

Dekan Fakultas Sains dan Teknologi



(Hamdani, ST., MT.)

Ketua Program Studi Sistem Komputer



(Eko Hariyanto, S.Kom., M.Kom)

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : LIANA
NPM : 1614370190
Prodi : Sistem Komputer
Konsentrasi : Keamanan Jaringan Komputer
Judul Skripsi : Rancang Bangun Aplikasi Self Productivity List dalam
Mengatur Aktivitas Harian Berbasis Android

Dengan ini menyatakan bahwa :

1. Tugas akhir/ skripsi saya bukan hasil plagiat.
2. Saya tidak akan menuntut perbaikan nilai indeks kumulatif (IPK) setelah ujian Sidang Meja Hijau.
3. Skripsi saya dapat dipublikasikan oleh lembaga, dan saya tidak akan menuntut akibat publikasi tersebut.

Demikian pernyataan ini saya perbuat dengan sebenar-benarnya, terimakasih.

Medan, 16 April 2021



LIANA

PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa dalam skripsi ini tidak terdapat karya yang diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis di acu dalam skripsi ini dan disebutkan dalam daftar pustaka.

Medan, 16 April 2021



LIANA



UNIVERSITAS PEMBANGUNAN PANCA BUDI FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO	(TERAKREDITASI)
PROGRAM STUDI ARSITEKTUR	(TERAKREDITASI)
PROGRAM STUDI SISTEM KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI TEKNIK KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI AGROTEKNOLOGI	(TERAKREDITASI)
PROGRAM STUDI PETERNAKAN	(TERAKREDITASI)

PERMOHONAN PRA PENGAJUAN TESIS / SKRIPSI / TUGAS AKHIR

Saya yang bertanda tangan di bawah ini :

Nama Lengkap : LIANA
Tempat/Tgl. Lahir : Sungal Ular / 08 Februari 1999
Nomor Pokok Mahasiswa : 1614370190
Program Studi : Sistem Komputer
Konsentrasi : Keamanan Jaringan Komputer
Jumlah Kredit yang telah dicapai : 141 SKS, IPK 3.70

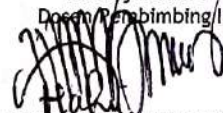
Dengan ini mengajukan judul skripsi sesuai dengan bidang ilmu, dengan judul:

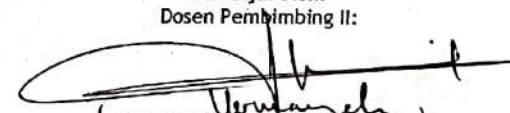
No.	Judul	Persetujuan
1	Rancang Bangun Aplikasi Notification Productivity Journal (Jurnal Produktivitas) Berbasis Android	<input checked="" type="checkbox"/> 28/3-20
2	Sistem Pakar Menentukan Aktivitas Keseharian yang Produktif diusia 17-60 Tahun dengan Metode Forward Chaining	<input type="checkbox"/>
3	Perancangan sistem pakar dalam menentukan kualitas tanaman padi dengan metode Forward Chaining	<input type="checkbox"/>

Medan, 04 Maret 2020

Pemohon,


(Liana)

Tanggal : 4 Maret 2020.....
Disetujui oleh :
Dosen Pembimbing I :

(.....)

Tanggal : 4 Maret 2020.....
Disetujui oleh :
Dosen Pembimbing II :

(.....)

No. Dokumen: FM-UPBM-18-01	Revisi: 0	Tgl. Eff: 22 Oktober 2018
----------------------------	-----------	---------------------------

**SURAT PERNYATAAN
PERUBAHAN JUDUL SKRIPSI**

Saya yang bertanda-tangan di bawah ini :

Nama : LIANA
NPM : 1614370190
Program Studi : Sistem Komputer

menyatakan **benar** bahwa judul skripsi saya mengalami perubahan sesuai dengan arahan dari dosen pembimbing saya. Judul skripsi saya pertama yang telah disetujui oleh Ketua Program Studi adalah :

“
.....
Rancang Bangun Aplikasi Notification Productivity Journal
(Jurnal Produktivitas) Berbasis Android
.....”

dan judul skripsi saat ini setelah diubah adalah :

“
.....
Rancang Bangun Aplikasi Self Productivity List
Dalam Mengatur Aktivitas Harian Berbasis Android
.....”

Demikian surat pernyataan ini saya perbuat dengan sebenar-benarnya.

Medan, 5 November 2020

Dibuat oleh,



LIANA

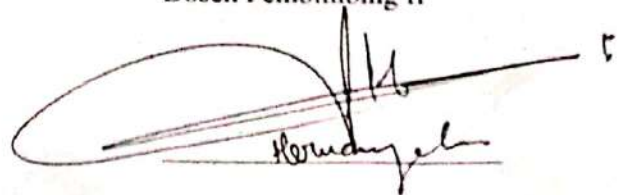
NPM. 1614370190

Diketahui oleh,

Dosen Pembimbing I



Dosen Pembimbing II


Herudyanah

UNIVERSITAS PEMBANGUNAN PANCA BUDI FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO
PROGRAM STUDI ARSITEKTUR
PROGRAM STUDI SISTEM KOMPUTER
PROGRAM STUDI TEKNIK KOMPUTER
PROGRAM STUDI AGROTEKNOLOGI
PROGRAM STUDI PETERNAKAN

(TERAKREDITASI)
(TERAKREDITASI)
(TERAKREDITASI)
(TERAKREDITASI)
(TERAKREDITASI)
(TERAKREDITASI)

PERMOHONAN JUDUL TESIS / SKRIPSI / TUGAS AKHIR*

Bertanda tangan di bawah ini :

Nama Lengkap :
Tgl. Lahir :
Jenis Kelamin :
No. Mahasiswa :
Bidang Studi :
Kredit yang telah dicapai :

: LIANA
: Sungai Ular / 08 Februari 1999
: 1614370190
: Sistem Komputer
: Keamanan Jaringan Komputer
: 141 SKS, IPK 3.70
: 082274430010


ingin mengajukan judul sesuai bidang ilmu sebagai berikut :

Judul

Rancang Bangun Aplikasi Self Productivity List dalam Mengatur Aktivitas Harian Berbasis Android


Disetujui Oleh Dosen Jika Ada Perubahan Judul

yang Tidak Perlu


 Rektor I,
 Cahyo Pramono, S.E., M.M.

Medan, 18 November 2020

Pemohon,


 (Liana)

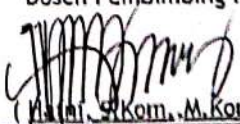
Tanggal :

Disahkan oleh :
Dekan


 (Hamdan, ST., MT)

Tanggal : 20 April 2021

Disetujui oleh :
Dosen Pembimbing I :


 (Hani, S.Kom., M.Kom.)

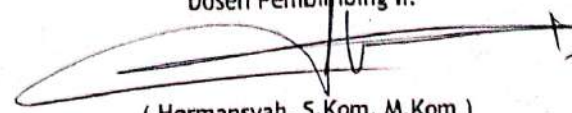
Tanggal :

Disetujui oleh:
Ka. Prodi Sistem Komputer


 (Eko Hariyanto, S.Kom., M.Kom.)

Tanggal :

Disetujui oleh:
Dosen Pembimbing II:


 (Hermansyah, S.Kom., M.Kom.)



YAYASAN PROF. DR. H. KADIRUN YAHYA

UNIVERSITAS PEMBANGUNAN PANCA BUDI

JL. Jend. Gatot Subroto KM 4,5 PO. BOX 1099 Telp. 061-30106057 Fax. (061) 4514808
MEDAN - INDONESIA

Website : www.pancabudi.ac.id - Email : admin@pancabudi.ac.id

LEMBAR BUKTI BIMBINGAN SKRIPSI

Nama Mahasiswa : LIANA
NPM : 1614370190
Program Studi : Sistem Komputer
Jenjang Pendidikan : Strata Satu
Dosen Pembimbing : Hafni, S.Kom.,M.Kom.
Judul Skripsi : Rancang Bangun Aplikasi Self Productivity List dalam Mengatur Aktivitas Harian Berbasis Android

Tanggal	Pembahasan Materi	Status	Keterangan
16 Mei 2020	Ass w w silakan upload bab pada portal ok	Revisi	
06 Juli 2020	Ass w w acc bab 2 lanjut ke bab 3	Disetujui	
20 Juli 2020	Ass w w pada bab 3 tidak kelihatan daftar aktifitasnya	Revisi	
20 Juli 2020	Ass w w 0% pada progress mingguan maksud apa	Revisi	
12 Agustus 2020	Ass w w acc bab 3 lanjut ke bab 4	Disetujui	
24 Agustus 2020	Ass w w acc bab 4 lanjut ke bab 5, persiapkan berkas untuk daftar seminar hasil	Disetujui	
21 Oktober 2020	Ass w w acc bab 5 siapkan berkas nya, dan buat video program sampai selesai, kirimkan ke wa	Disetujui	
21 Oktober 2020	Ass w w lengkapi berkas untuk mendaftar seminar hasil	Revisi	
21 Oktober 2020	Ass w w acc bab 5, kemudian lengkapi berkas untuk mendaftar seminar hasil, buat video menjalankan program kirimkan ke wa ok	Disetujui	
27 Oktober 2020	Ass w w acc seminar hasil	Disetujui	
15 Desember 2020	Ass w w Acc sidang Meja hijau	Disetujui	
18 Februari 2021	Ass w w Acc jilid	Disetujui	

Medan, 15 April 2021
Dosen Pembimbing,



Hafni, S.Kom.,M.Kom.



YAYASAN PROF. DR. H. KADIRUN YAHYA

UNIVERSITAS PEMBANGUNAN PANCA BUDI

JL. Jend. Gatot Subroto KM 4,5 PO. BOX 1099 Telp. 061-30106057 Fax. (061) 4514808
MEDAN - INDONESIA

Website : www.pancabudi.ac.id - Email : admin@pancabudi.ac.id

LEMBAR BUKTI BIMBINGAN SKRIPSI

Nama Mahasiswa : LIANA
NPM : 1614370190
Program Studi : Sistem Komputer
Jenjang Pendidikan : Strata Satu
Dosen Pembimbing : Hermansyah, S.Kom, M.Kom
Judul Skripsi : Rancang Bangun Aplikasi Self Productivity List dalam Mengatur Aktivitas Harian Berbasis Android

Tanggal	Pembahasan Materi	Status	Keterangan
01 Juli 2020	BAB II: - Meringkan semua Tulisan Asing, - gunakan MENDELEY untuk penulisan kutipan / referensi,	Revisi	
08 Juli 2020	Acc bab 2 lanjut BAB 3 dan Demo Program (Videokan saja bagaimana menjalankan Programnya)	Disetujui	
27 Juli 2020	Perbaiki BAB 3 : dalam Tahapan PEnelitian sama MEtode Penelitian ...! mana yang mau dipakai? isi nya hampir sama. bedanya 3.2 metode nya sampai perawatan. pilih salah satu aja biar ga double.	Revisi	
13 Agustus 2020	Lanjut BAB 4	Revisi	
11 September 2020	PERbaiki dulu bab 3 sesuai buku panduan penulisan skripsi yang baru	Revisi	
11 September 2020	Perbaiki Bab 4 maksudnya sesuai buku panduan skripsi yang baru	Revisi	
23 Oktober 2020	Lengkapi Semua. ACC Seminar HASil	Disetujui	
15 Desember 2020	Acc Sidang Meja Hijau	Disetujui	
19 Februari 2021	Acc Jilid	Disetujui	

Medan, 15 April 2021
Dosen Pembimbing,



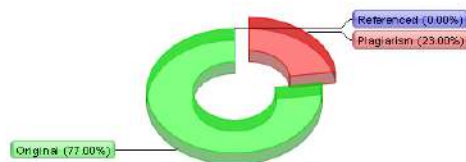
Hermansyah, S.Kom, M.Kom

Plagiarism Detector v. 1460 - Originality Report 16-Dec-20 16:14:19

Analyzed document: LIANA_1614370190_SISTEM KOMPUTER.docx Licensed to: Universitas Pembangunan Panca Budi_License03

Comparison Preset: Rewrite. Detected language: Indonesian

Relation chart:



Distribution graph:



Top sources of plagiarism:

	% 11		https://repository.bsi.ac.id/index.php/unduh/item/2672/Tugas-Akhir-Perancangan-S...
	% 10		https://repository.bsi.ac.id/index.php/unduh/item/287524/14.-BAB-11-Landasan-Teo...
	% 4		https://kosemsme.blogspot.com/2015/04/use-case-diagram.html#:~:text=Relas%20us...

[Show other Sources.]

Processed resources details:

[Show other Sources:]	94 - OK / 42 - Failed
-----------------------	-----------------------

Processed resources:

SURAT KETERANGAN PLAGIAT CHECKER

Dengan ini saya Ka.LPMU UNPAB menerangkan bahwa surat ini adalah bukti pengesahan dari LPMU sebagai pengesah proses plagiat checker Tugas Akhir/ Skripsi/Tesis selama masa pandemi *Covid-19* sesuai dengan edaran rektor Nomor : 7594/13/R/2020 Tentang Pemberitahuan Perpanjangan PBM Online.

Demikian disampaikan.

NB: Segala penyalahgunaan/pelanggaran atas surat ini akan di proses sesuai ketentuan yang berlaku UNPAB.



No. Dokumen : PM-UJMA-06-02	Revisi : 00	Tgl Eff : 23 Jan 2019
-----------------------------	-------------	-----------------------



KARTU BEBAS PRAKTIKUM
Nomor. 1060/BL/LAKO/2020

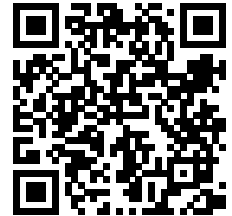
Yang bertanda tangan dibawah ini Ka. Laboratorium Komputer dengan ini menerangkan bahwa :

Nama : LIANA
N.P.M. : 1614370190
Tingkat/Semester : Akhir
Fakultas : SAINS & TEKNOLOGI
Jurusan/Prodi : Sistem Komputer

Benar dan telah menyelesaikan urusan administrasi di Laboratorium Komputer Universitas Pembangunan Panca Budi Medan.

Medan, 17 Desember 2020
Ka. Laboratorium

Melva Sari Panjaitan, S. Kom., M.Kom.





YAYASAN PROF. DR. H. KADIRUN YAHYA
PERPUSTAKAAN UNIVERSITAS PEMBANGUNAN PANCA BUDI
Jl. Jend. Gatot Subroto KM. 4,5 Medan Sunggal, Kota Medan Kode Pos 20122

SURAT BEBAS PUSTAKA
NOMOR: 3418/PERP/BP/2020

Kepala Perpustakaan Universitas Pembangunan Panca Budi menerangkan bahwa berdasarkan data pengguna perpustakaan atas nama saudara/i:

Nama : LIANA
N.P.M. : 1614370190
Tingkat/Semester : Akhir
Fakultas : SAINS & TEKNOLOGI
Jurusan/Prodi : Sistem Komputer

Bahwasannya terhitung sejak tanggal 17 Desember 2020, dinyatakan tidak memiliki tanggungan dan atau pinjaman buku sekaligus tidak lagi terdaftar sebagai anggota Perpustakaan Universitas Pembangunan Panca Budi Medan.

Medan, 17 Desember 2020
Diketahui oleh,
Kepala Perpustakaan,



Sugiarjo, S.Sos., S.Pd.I

Hal : Permohonan Meja Hijau

Medan, 17 Desember 2020
Kepada Yth : Bapak/Ibu Dekan
Fakultas SAINS & TEKNOLOGI
UNPAB Medan
Di -
Tempat

Dengan hormat, saya yang bertanda tangan di bawah ini :

Nama : LIANA
Tempat/Tgl. Lahir : Sungai Ular / 8 Februari 1999
Nama Orang Tua : TIBRANI
N. P. M : 1614370190
Fakultas : SAINS & TEKNOLOGI
Program Studi : Sistem Komputer
No. HP : 082274430010
Alamat : DUSUN 1 SUNGAI ULAR

Datang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **Rancang Bangun Aplikasi Self Productivity List dalam Mengatur Aktivitas Harian Berbasis Android**, Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntut ujian perbaikan nilai mata kuliah untuk perbaikan indek prestasi (IP), dan mohon diterbitkan ijazahnya setelah lulus ujian meja hijau.
3. Telah tercap keterangan bebas pustaka
4. Terlampir surat keterangan bebas laboratorium
5. Terlampir pas photo untuk ijazah ukuran 4x6 = 5 lembar dan 3x4 = 5 lembar Hitam Putih
6. Terlampir foto copy STTB SLTA dilegalisir 1 (satu) lembar dan bagi mahasiswa yang lanjutan D3 ke S1 lampirkan ijazah dan transkripnya sebanyak 1 lembar.
7. Terlampir pelunasan kwintasi pembayaran uang kuliah berjalan dan wisuda sebanyak 1 lembar
8. Skripsi sudah dijilid lux 2 exemplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jilid kertas jeruk 5 exemplar untuk penguji (bentuk dan warna penjiilidan diserahkan berdasarkan ketentuan fakultas yang berlaku) dan lembar persetujuan sudah di tandatangani dosen pembimbing, prodi dan dekan
9. Soft Copy Skripsi disimpan di CD sebanyak 2 disc (Sesuai dengan Judul Skripsinya)
10. Terlampir surat keterangan BKKOL (pada saat pengambilan ijazah)
11. Setelah menyelesaikan persyaratan point-point diatas berkas di masukan kedalam MAP
12. Bersedia melunaskan biaya-biaya uang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan perincian sbb :

1. [102] Ujian Meja Hijau	: Rp.	0
2. [170] Administrasi Wisuda	: Rp.	1,500,000
3. [202] Bebas Pustaka	: Rp.	100,000
4. [221] Bebas LAB	: Rp.	5,000
Total Biaya	: Rp.	1,605,000

Ukuran Toga :

S

Diketahui/Disetujui oleh :

Hormat saya



Hamdani, ST., MT.
Dekan Fakultas SAINS & TEKNOLOGI



LIANA
1614370190

Catatan :

- 1. Surat permohonan ini sah dan berlaku bila ;
 - a. Telah dicap Bukti Pelunasan dari UPT Perpustakaan UNPAB Medan.
 - b. Melampirkan Bukti Pembayaran Uang Kuliah aktif semester berjalan
- 2. Dibuat Rangkap 3 (tiga), untuk - Fakultas - untuk BPAA (asli) - Mhs.ybs.

ABSTRAK

LIANA

Rancang Bangun Aplikasi *Self Productivity List* Dalam Mengatur Aktivitas Harian Berbasis Android 2021

Produktivitas merupakan suatu pengetahuan dan keterampilan yang diperlukan untuk seseorang bisa meningkatkan output dari kehidupannya dalam mencapai apa yang penting baginya dari usaha yang dilakukan. Salah satu cara untuk menjadi produktif yaitu dengan melakukan suatu perencanaan mulai dengan membuat *To Do List* menggunakan media buku atau kertas catatan. Seiring dengan tingkat mobilitas yang tinggi, *smartphone* yang awalnya hanya sebagai media komunikasi sekarang bisa melebihi fungsi dasarnya seperti membantu dalam mengatur aktivitas harian. Aplikasi *Self Productivity List* lebih dari sekedar membuat *To Do List*. Aplikasi ini dapat memberikan notifikasi pada *To Do List* yang dibuat dan juga dilengkapi dengan fitur-fitur lainnya seperti fitur List untuk melihat daftar perencanaan yang sudah dibuat, Kalender, Progres untuk melihat rangkuman pencapaian yang telah dilakukan dari perencanaan yang telah dibuat, Pengaturan, dan Tantangan untuk menciptakan kebiasaan yang dilakukan selama 30 hari berturut-turut.

Kata Kunci : Aplikasi *Self Productivity List*, Produktivitas, *Smartphone*, *To Do List*.

KATA PENGANTAR

Puji dan syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya yang selalu dilimpahkan kepada hambanya sehingga dapat menyelesaikan penelitian dan skripsi ini dengan judul: **“RANCANG BANGUN APLIKASI *SELF PRODUCTIVITY LIST* DALAM MENGATUR AKTIVITAS HARIAN BERBASIS ANDROID”**.

Dalam penyusunan skripsi ini tidak terlepas dari berbagai bantuan dan dukungan dari berbagai pihak. Pada kesempatan ini penulis mengucapkan terimakasih kepada :

1. Orang tua tercinta yang telah mengasuh dan membesarkan penulis dengan rasa cinta dan kasih sayang, yang telah memberikan dukungan dan do'a selama penulis menjalani pendidikan.
2. Bapak Dr. H. Muhammad Isa Indrawan, S.E, M.M. selaku Rektor Universitas Pembangunan Panca Budi Medan.
3. Bapak Hamdani, ST., MT. selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
4. Bapak Eko Hariyanto, S.Kom., M.Kom. selaku Kepala Program Studi Sistem Komputer.
5. Bapak Hafni, S.Kom., M.Kom. selaku Dosen Pembimbing I yang telah memberikan bimbingan dan arahan selama proses pembuatan skripsi ini.

6. Bapak Hermansyah, S.Kom., M.Kom. selaku Dosen Pembimbing II yang telah memberikan bimbingan dan arahan selama proses pembuatan skripsi ini.
7. Seluruh Bapak/ Ibu dosen Program Studi Sistem Komputer yang telah memberikan ilmu yang sangat bermanfaat untuk penulis.
8. Teman-teman dan semua pihak yang telah mendukung dan membantu penulis dalam penyusunan skripsi ini.

Penulis menyadari sepenuhnya bahwa dalam penyusunan skripsi ini masih ada kekurangan, untuk itu diharapkan adanya masukan dan saran terutama dari bapak/ ibu pembimbing serta rekan-rekan mahasiswa demi untuk menyempurnakan penulisan skripsi ini serta mengembangkannya di waktu mendatang. Akhir kata penulis ucapkan teimakasih.

Medan, Februari 2021

LIANA

DAFTAR ISI

ABSTRAK	i
KATA PENGANTAR	ii
DAFTAR ISI	iv
DAFTAR TABEL	vii
DAFTAR GAMBAR	viii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	5
BAB II LANDASAN TEORI	6
2.1 Penelitian Terdahulu	6
2.2 Produktivitas Diri	8
2.3 Aplikasi	11
2.4 Android	13
2.4.1 Versi Android	13
2.4.2 Fitur-fitur Android	15
2.5 C#	17
2.6 Visua Studio	18
2.7 UML (<i>Unified Modelling Language</i>)	19

2.7.1	<i>Use Case Diagram</i>	20
2.7.2	<i>Activity Diagram</i>	22
2.7.3	<i>Sequence Diagram</i>	23
2.7.4	<i>Class Diagram</i>	25
BAB III METODE PENELITIAN		27
3.1	Tahapan Penelitian	27
3.2	Metodologi Penelitian	29
3.2.1	Metode Pengembangan Perangkat Lunak	29
3.3	Analisis Sistem	31
3.4	Analisis Kebutuhan	32
3.4.1	Analisis Kebutuhan Fungsionalitas	32
3.5	Perancangan Arsitektur Navigasi	34
3.6	Perancangan Sistem	34
3.6.1	<i>Use Case Diagram</i>	34
3.6.2	<i>Activity Diagram</i>	38
3.6.3	<i>Sequence Diagram</i>	40
3.6.4	<i>Class Diagram</i>	43
3.6.5	Perancangan Tabel	43
3.6.6	Perancangan Antarmuka (<i>User Interface</i>)	44
BAB IV HASIL DAN PEMBAHASAN		51
4.1	Implementasi Sistem Yang Digunakan	51
4.2	Implementasi <i>Interface</i>	52
4.3	Pengujian Aplikasi <i>Self Productivity List</i>	62

4.4 Pengujian Kuesioner	66
BAB V PENUTUP	69
5.1 Kesimpulan	69
5.2 Saran	70
DAFTAR PUSTAKA	71
BIOGRAFI PENULIS	
LAMPIRAN	

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Berbicara tentang produktivitas memang tidak akan ada habisnya. Apalagi seiring dengan perkembangan zaman, semakin besar tantangan yang dihadapi untuk membuat hidup menjadi produktif. Seringkali dengan berbagai perkembangan teknologi yang ada, bukannya menjadi produktif dan bisa memanfaatkan waktu dengan baik, justru yang terjadi malah mengganggu aktivitas utama. Produktivitas merupakan suatu pengetahuan dan keterampilan yang diperlukan untuk seseorang bisa meningkatkan output dari kehidupannya dalam mencapai apa yang penting baginya dari usaha yang dilakukan. Ini adalah salah satu soft skill penting yang perlu dimiliki setiap orang. Produktivitas merupakan sikap mental yang mempunyai pandangan bahwa untuk kehidupan hari ini akan lebih baik dari hari kemarin dan hari esok adalah lebih baik dari hari ini.

Produktivitas yaitu tentang membuat pilihan yang cerdas secara konsisten dengan energi, fokus, dan waktu untuk memaksimalkan potensi serta meraih hasil yang bermanfaat (Faris, 2017). Membuat niat implementasi adalah strategi yang dapat digunakan untuk memasang kebiasaan produktif baru dengan waktu dan lokasi tertentu (Clear, 2018). Untuk membangun kebiasaan produktif tentunya tidak bisa didapatkan secara instan. Produktivitas adalah proses. Perubahan sebenarnya dibangun atas kebiasaan-kebiasaan kecil yang dibuat setiap hari yang

membantu seseorang menjadi produktif (G.Suardhika 2015). Kita menciptakan kebiasaan pada kali pertama, selanjutnya kebiasaanlah yang akan menciptakan kita (Abduh, 2018). Ada banyak cara untuk menjadi produktif, salah satunya yaitu dengan melakukan suatu perencanaan baik untuk tujuan perbaikan atau peningkatan maupun tujuan pengembangan. Pada umumnya untuk memulai suatu perencanaan seseorang biasanya membuat semacam *To Do List* (daftar yang harus dilakukan) pada buku atau kertas catatan yang kemudian ditempelkan pada dinding ataupun tempat-tempat semacamnya agar mudah diingat. Namun membuat *To Do List* secara manual tidak begitu efektif bagi para penggunanya. *To Do List* yang dibuat dengan media buku atau kertas catatan (seperti *sticky note*) sangat mudah dilupakan karena hanya berada disatu tempat tertentu dan tidak memiliki semacam tanda ataupun pemberitahuan untuk mudah dilihat dan diingat. Seiring dengan perkembangan zaman dan tingkat mobilitas yang tinggi *smartphone* yang awalnya hanya sebagai media komunikasi sekarang bisa melebihi fungsi dasarnya, berbagai macam fitur sudah bisa dirasakan di berbagai *smartphone* yang ada saat ini dari kamera, video, dokumen, pengolah data dan lain sebagainya. Hal ini tak lepas dari penggunaan sistem operasi pada *smartphone*. Layaknya komputer, *smartphone* saat ini dapat diinstal dengan berbagai macam program yang diinginkan. Android merupakan salah satu sistem operasi yang sangat diminati saat ini. Android mempunyai kelebihan di bidang pengembangan aplikasi, dengan sistem *open source* program android dapat dibuat oleh siapa saja dengan gratis. Android mempunyai tujuan utama untuk memajukan inovasi piranti telepon bergerak agar pengguna mampu

mengeksplorasi kemampuan dan menambah pengalaman lebih dibandingkan dengan *platform mobile* lainnya. Hingga saat ini Android mampu berkembang lebih pesat dari para pesaingnya dari segi sistem maupun aplikasinya oleh karena itu banyak aplikasi-aplikasi yang diciptakan dengan berbasis Android.

Dengan melihat dan mengamati sistem *To Do List* yang masih dilakukan secara manual, penulis termotivasi untuk merancang sebuah aplikasi *To Do List* pada *smartphone* berbasis android dengan nama *Self Productivity List*. Aplikasi tersebut akan menyediakan fitur-fitur yang bisa meningkatkan produktivitas seseorang untuk mengembangkan potensi diri serta meraih hasil yang bermanfaat, dimulai dengan mengatur aktivitas harian dan kegiatan-kegiatan sederhana lainnya. Berdasarkan hal-hal yang telah diuraikan tersebut, dalam penyusunan skripsi ini penulis memilih judul **“RANCANG BANGUN APLIKASI *SELF PRODUCTIVITY LIST* DALAM MENGATUR AKTIVITAS HARIAN BERBASIS ANDROID”**.

1.2 Rumusan Masalah

Dengan mengacu pada latar belakang masalah yang telah diuraikan tersebut, maka disusun rumusan masalah yang akan dibahas dalam skripsi ini adalah sebagai berikut :

1. Bagaimana merancang bangun aplikasi *Self Productivity List* dalam mengatur aktivitas harian berbasis Android?

2. Bagaimana merancang sebuah aplikasi pada *platform* android yang dapat digunakan dimana saja untuk membantu *user* dalam membuat suatu catatan perencanaan (*To Do List*).

1.3 Batasan Masalah

Agar pembahasan dalam penyusunan skripsi ini lebih terarah, maka penulis memberikan batasan masalah yaitu :

1. Perancangan aplikasi *Self Productivity List* ini hanya didukung pada perangkat *mobile* berbasis Android.
2. Aplikasi ini berfokus untuk mengukur tingkat produktivitas diri melalui aktivitas harian dan kegiatan-kegiatan lainnya.
3. Aplikasi ini menyediakan fitur produktivitas spiritual yang hanya diperuntukan bagi muslim, untuk membantu melakukan evaluasi ibadah harian.
4. Aplikasi bersifat *offline*.

1.4 Tujuan Penelitian

Berikut ini adalah beberapa tujuan penelitian yang akan dibahas dalam penelitian ini yaitu sebagai berikut :

1. Untuk merancang bangun aplikasi *Self Productivity List* dalam mengatur aktivitas harian berbasis Android.

2. Untuk memberikan pemahaman produktivitas diri yang baik dengan mengelola diri melalui kebiasaan-kebiasaan produktif yang tepat.
3. Untuk merancang sebuah aplikasi pada android yang dapat digunakan dimana saja untuk membuat suatu catatan perencanaan (*To Do List*).

1.5 Manfaat Penelitian

Berikut ini adalah beberapa manfaat penelitian yang akan dibahas dalam penelitian ini yaitu sebagai berikut :

1. Aplikasi ini memudahkan penggunanya untuk mengukur dan meningkatkan potensi diri melalui kebiasaan produktif yang tepat.
2. Mendapatkan sistem yang lebih praktis, efisien, dan dapat berjalan secara terintegrasi melalui aplikasi pada *platform* android.
3. Menjadikan teknologi khususnya *smartphone* sebagai dampak positif bagi setiap penggunanya terhadap penunjang kebutuhan dalam penyajian informasi.

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu menjadi hal yang sangat dibutuhkan bagi penulis sebagai data pendukung dan bahan pertimbangan yang terkait dengan penelitian yang sedang dilakukan. Penelitian terdahulu sangat dibutuhkan bagi peneliti sebagai referensi dalam memperkuat penelitian.

Penelitian yang dilakukan oleh Agrista Sarfina Shani tahun 2017, dengan judul Sistem Aplikasi *Water Reminder* Berbasis Android. Metode penelitian yang digunakan pada penelitian ini adalah metode SDLC (*System Development Life Cycle*). Hasil dari penelitian ini yaitu Aplikasi *Water Reminder* digunakan untuk mengontrol asupan air minum yang masuk ke dalam tubuh. Aplikasi *Water Reminder* ini berisi fitur dan menu edukasi tentang pentingnya mengonsumsi air bagi tubuh, menghitung target harian dan mingguan dalam mengonsumsi air, mengatur alarm sebagai pengingat minum air, dan memantau air yang sudah berhasil diminum selama satu hari hingga satu minggu kemudian.

Penelitian yang dilakukan oleh Belen Septian, Ghiri Basuki Putra, dan Fardhan Arkan tahun 2016, dengan judul Rancang Bangun Aplikasi Pengingat Aktivitas Perkuliahan Berbasis Android. Penelitian ini menggunakan aplikasi Eclipse untuk membuat aplikasi android dan Sqlite dan MySQL sebagai basis data aplikasinya. Hasil dari penelitian ini yaitu aplikasi ini bertujuan untuk

memudahkan mahasiswa dalam mengingat aktivitas perkuliahan dengan tepat waktu serta membantu mahasiswa mendapatkan notifikasi secara *real time* dan *online* dari kampus.

Penelitian yang dilakukan oleh Devi Dwi Purwanto tahun 2014, dengan judul Aplikasi *Reminder* Pada *Android Mobile Device*. Penelitian ini melakukan analisa referensi aplikasi sejenis, analisa kebutuhan sistem, solusi yang ditawarkan, teknologi yang digunakan, hingga menjelaskan deskripsi sistem yang baru. Hasil dari penelitian ini yaitu Aplikasi *Reminder* digunakan untuk membuat pencatatan kegiatan yang dapat memberikan pengingat untuk setiap kegiatan dan juga mampu membantu mengingatkan para pihak yang juga ikut serta dalam kegiatan tersebut. Aplikasi ini juga dilengkapi dengan fitur yang dapat menunjukkan lokasi dari kegiatan yang telah disimpan sebelumnya dalam peta oleh pengguna.

Penelitian yang dilakukan oleh A.Astuty Indriani, Ratnawati, dan First Wanita tahun 2019, dengan judul *Reminder* Pengontrolan Perawatan Gigi Berbasis Android. Metode penelitian yang digunakan pada penelitian ini yaitu metode observasi (dokter spesialis gigi, buku referensi android, jurnal, *paper*) dan metode analisis (analisis data, analisis sistem, dan analisis pengujian). Hasil rancangan aplikasi ini memudahkan dokter dan pasien untuk melihat *history* pemeriksaan gigi, mengingatkan jadwal pengontrolan perawatan gigi pada dokter dan pasien, serta mendapatkan informasi biaya perawatan gigi pada klinik Goa Ria.

Dalam penelitian-penelitian yang telah dilakukan di atas, penulis menemukan bahwa semua aplikasi yang dibuat berbasis android. Hal ini membuktikan bahwa *smartphone* android lebih banyak digunakan oleh masyarakat dibandingkan dengan *smartphone* lainnya. Hal itu juga disebabkan akan keunggulan android yang bersifat *open source*. Semua aplikasi yang dirancang pada penelitian-penelitian tersebut memiliki tujuan yang sama, yaitu mengingatkan penggunaannya akan kegiatan yang telah dijadwalkan.

Pada penelitian ini penulis akan membuat aplikasi pengingat yang bersifat umum, dapat digunakan untuk mengatur aktivitas/ kegiatan apapun, tidak dikhususkan pada satu aktivitas/ kegiatan tertentu. Aplikasi yang akan dirancang berfokus untuk meningkatkan produktivitas pengguna dengan mengatur aktivitas/ kegiatan yang dilakukan.

2.2 Produktivitas Diri

Produktivitas diri (kata 'diri' ditambahkan untuk membedakannya dengan produktivitas organisasi) pada umumnya didefinisikan sebagai output dibagi dengan input, yang mana diukur dengan berapa banyak yang dapat dilakukan dan dicapai seseorang dalam suatu periode tertentu (Suardhika, 2015).

Menurut (Faris, 2017), ada empat mitos tentang produktivitas, diantaranya adalah sebagai berikut :

1. Produktivitas bukan berarti sibuk

Sibuk sepanjang hari tetapi mungkin tidak produktif. Kesibukan tersebut hanya membuang-buang energi, fokus, dan waktu dalam aktivitas yang tak ada artinya yang tidak memberikan nilai tambah bagi kehidupan dan mencapai tujuan. Mohammed Faris berpendapat bahwa orang yang produktif seharusnya tidak terlalu sibuk dan tidak terlihat stress.

2. Produktivitas bukanlah sebuah kejadian

Produktivitas adalah sebuah proses, butuh waktu untuk menjadi seorang yang produktif. Produktivitas adalah tentang membuat pilihan cerdas setiap hari hingga kebiasaan-kebiasaan yang bermanfaat tertanam dan sikap produktif menjadi gaya hidup.

3. Produktivitas itu tidak membosankan

Menjadi produktif bukan berarti tidak lagi menonton TV, membuka Facebook, pergi jalan-jalan bersama teman-teman atau bersenang-senang. Menjadi produktif berarti mengetahui kapan harus bersenang-senang dan kapan harus bekerja keras, kapan bersikap rileks dan kapan bersikap serius. Produktif adalah membuat pilihan-pilihan cerdas.

4. Anda tidak bisa selalu menjadi produktif

Salah satu tantangan yang dihadapi tentang produktivitas adalah mempertahankan rutinitas produktif yang konsisten. Beberapa orang mengeluh bahwa mereka akan menjadi orang yang produktif selama satu pekan, kemudian

bermalas-malasan selama dua pekan berikutnya. Mohammed Faris memberitahu bahwa walaupun ada banyak cara untuk mempertahankan tingkat produktivitas dalam hari/ kehidupan, jangan pernah berpikir bahwa kita adalah sebuah mesin yang dapat bekerja secara konsisten dengan kecepatan dan tingkat produktivitas yang tinggi. Bahkan mesin sekalipun akan rusak jika selalu bergerak pada kecepatan tinggi.

(Faris, 2017), membuat Istilah “produktivitas spiritual” sebagai cara untuk menggambarkan bagaimana spiritualitas dapat meningkatkan produktivitas kita. Memahami produktivitas spiritual bertujuan untuk memahami kaitan antara dunia dan akhirat.

Dalam islam produktivitas dipandang sebagai alat, bukan sebagai tujuan itu sendiri. Produktivitas menyediakan tujuan dan seperangkat nilai bagi kita untuk hidup di dunia ini. Di samping itu, ia memelihara jiwa kita sehingga keseimbangan antara tubuh, pikiran, dan jiwa terpelihara dalam kehidupan kita sehari-hari.

Produktivitas diri seseorang diukur dengan berapa banyak yang dapat dilakukannya dan capai dalam suatu periode tertentu. Produktivitas diri adalah salah satu elemen kunci dalam mengelola hidup. Jika seseorang dapat mengelola hidup ia akan dapat hidup secara lebih bermakna.

2.3 Aplikasi

Aplikasi adalah suatu sistem yang dirancang untuk mengolah data dengan aturan serta ketentuan dengan menggunakan bahasa pemrograman tertentu. Aplikasi bisa dikatakan suatu perangkat lunak (*software*) yang siap pakai dengan menjalankan instruksi-instruksi dari pengguna (*user*). Aplikasi banyak diciptakan untuk membantu berbagai keperluan seperti membuat laporan, percetakan dan lain-lain.

Adapun pengertian aplikasi menurut para ahli adalah sebagai berikut:

1. Menurut Jogiyanto (1999), aplikasi adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*.
2. Menurut Hengky W. Pramana, aplikasi merupakan suatu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, *game* pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia.
3. Menurut Eko. I dan Djokopran, aplikasi merupakan proses atau prosedur aliran data dalam infrastruktur teknologi informasi yang dapat dimanfaatkan oleh para pengambil keputusan yang sesuai dengan jenjang dan kebutuhan (relevan).

Secara sederhana, aplikasi dapat diartikan sebagai suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia.

Adapun klasifikasi aplikasi adalah sebagai berikut:

1. *System Software*, yaitu aplikasi yang dapat mengelola dan mengendalikan proses operasi *internal* di dalam sistem komputer.
2. *Real Time Software*, yaitu aplikasi yang memiliki fungsi untuk mengamati, menganalisa, serta mengendalikan suatu kejadian di dunia nyata saat sedang berlangsung.
3. *Business Software*, yaitu aplikasi yang diciptakan dan dikembangkan untuk keperluan bisnis, misalnya untuk mengatur sistem keuangan.
4. *Engineering and Scientific Software*, yaitu aplikasi yang diciptakan dan dikembangkan untuk membantu manusia menyelesaikan suatu masalah yang sifatnya non algoritmik.
5. *Web Based Software*, yaitu aplikasi yang memiliki fungsi sebagai media penghubung antara pengguna dengan internet secara langsung.
6. *Personal Computer Software*, yaitu aplikasi yang digunakan untuk perangkat pengguna resmi dan juga pribadi yang sudah banyak digunakan saat ini.

2.4 Android

Android adalah sistem operasi *mobile* yang bersifat *open source*, yang artinya android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan suatu aplikasi sendiri yang mampu berjalan diatas piranti android. Pada awalnya android dikembangkan oleh Android Inc dan kemudian beralih sepenuhnya ke tangan *Google Corporation* yang dibeli pada tahun 2005.

Ada berbagai macam alasan mengapa aplikasi berbasis android perlu dikembangkan, yakni untuk menjawab kebutuhan bisnis, membangun layanan baru, membuat bisnis baru, dan menyediakan *game* serta jenis materi lainnya untuk pengguna, dan alasan yang lebih utama mengembangkan aplikasi android dari *developer* adalah agar dapat menjangkau sebagian besar pengguna perangkat seluler dalam berbagai bidang (Herlina & Musliadi KH, 2019).

Android menjadi sistem operasi yang sangat populer karena tingkat efektivitas dan efisiensinya yang lebih baik dibandingkan dengan program sejenisnya seperti *Mobile Window* atau sistem operasi Symbian (Ismayani, 2018).

2.4.1 Versi Android

Dikutip dari Wikipedia, berikut adalah versi android dari pertama sampai skripsi ini ditulis :

Tabel 2.1 Versi Android

Versi	Nama Kode	Tanggal Rilis	Level API
1.0	N/A	23 September 2008	1
1.1	N/A	9 Februari 2009	2
1.5	<i>Cupcake</i>	30 April 2009	3
1.6	<i>Donut</i>	15 September 2009	4
2.0	<i>Eclair</i>	26 Oktober 2009	5
2.0.1	<i>Eclair</i>	3 Desember 2009	6
2.1	<i>Eclair</i>	12 Januari 2010	7
2.2-2.2.3	<i>Froyo</i>	20 Mei 2010	8
2.3-2.3.2	<i>Gingerbread</i>	6 Desember 2010	9
2.3.3-2.3.7	<i>Gingerbread</i>	9 Februari 2011	10
3.0	<i>Honeycomb</i>	22 Februari 2011	11
3.1	<i>Honeycomb</i>	10 Mei 2011	12
3.2	<i>Honeycomb</i>	15 Juli 2011	13
4.0-4.0.2	<i>Ice Cream Sandwich</i>	19 Oktober 2011	14
4.0.3-4.0.4	<i>Ice Cream Sandwich</i>	16 Desember 2011	15
4.1	<i>Jelly Bean</i>	9 Juli 2012	16
4.2	<i>Jelly Bean</i>	13 November 2012	17
4.3	<i>Jelly Bean</i>	24 Juli 2013	18
4.4	<i>KitKat</i>	31 Oktober 2013	19
5.0	<i>Lolipop</i>	25 Juni 2014	21

6.0	<i>Marshmallow</i>	5 Oktober 2015	23
7.0	<i>Nougat</i>	22 Agustus 2016	24
7.1	<i>Nougat</i>	4 Oktober 2016	25
8.0	<i>Oreo</i>	21 Agustus 2017	26
8.1.0	<i>Oreo</i>	5 Desember 2017	27
9	<i>Pie</i>	6 Agustus 2018	28
10	<i>Q</i>	3 September 2019	29

2.4.2 Fitur-fitur Android

Berikut ini adalah beberapa fitur yang dimiliki oleh android, diantaranya:

1. *User interface (UI)*

Sistem operasi android memiliki UI menarik, interaktif dan intuitif bagi penggunanya.

2. *Konektifitas (Connectivity)*

Sistem operasi android mendukung konektifitas seperti GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, *Bluetooth*, Wi-Fi, LTE, NFC dan WiMAX.

3. *Storage*

Menggunakan SQLite sebagai *database* relasional.

4. *Media Support*

Mendukung perangkat-perangkat media seperti suara, citra, video (H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC,

AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, dan BMP).

5. *Messaging*

SMS dan MMS.

6. *Web Browser*

Memiliki *web browser Chrome V-* yang berbasis Javascript dan mendukung HTML 5 dan CSS3.

7. *Multi-Touch*

Memiliki fitur bawaan *multi touch*.

8. *Multi-Tasking*

Fitur yang mengizinkan pengguna untuk menjalankan beberapa aplikasi secara bersama.

9. *Cosutume Widget*

Memiliki *widget-widget* yang bisa diubah-ubah.

10. *Multi-Language*

Memiliki fitur yang mendukung berbagai bahasa di dunia.

11. *Google Cloud Messaging (GCM)*

GCM merupakan layanan yang mengizinkan pengembang untuk mengirimkan pesan singkat/ data ke pengguna perangkat android.

12. *Wi-Fi*

Mendukung fitur Wi-Fi yang terdapat pada perangkat *mobile*.

13. *Android Beam*

Merupakan fitur berbasis teknologi NFC yang digunakan untuk berbagi secara instan.

2.5 C

C# atau *C-Sharp* adalah suatu bahasa pemrograman berorientasi objek yang dikembangkan oleh Microsoft yang dipimpin oleh Anders Hejlsberg. C# dianggap sebagai kombinasi antara efisiensi pemrograman C++, kesederhanaan pemrograman java, dan penyederhanaan dari pemrograman Visual Basic (Hakim, 2018).

Dikutip dari Wikipedia, C# didisain untuk memenuhi kebutuhan akan sintaksis C++ yang lebih ringkas dan *Rapid Application Development* yang 'tanpa batas' (dibandingkan dengan RAD yang 'terbatas' seperti yang terdapat pada Delphi dan Visual Basic).

Secara sederhana, C# merupakan bahasa pemrograman sederhana berorientasi objek yang dikembangkan dari bahasa pendahulunya, yaitu C dan C++.

Menurut (Darmawan, 2015), menjelaskan ada beberapa hal yang menjadi kelebihan dari bahasa pemrograman C# dibandingkan dengan bahasa pemrograman lainnya, yaitu :

1. Termasuk bahasa pemrograman .NET, dengan demikian kita dapat menggunakan komponen-komponen yang dibangun dengan bahasa pemrograman .NET lainnya.

2. Bahasa Pemrograman C# memiliki *Language Integrated Query* (LINQ) yang merupakan sintaks *query* yang dapat digunakan pada setiap kumpulan data.
3. *Windows Presentation Foundation* (WPF) dapat digunakan untuk membuat tampilan aplikasi kita dengan sangat kreatif.

Microsoft memberikan IDE (*Software* yang digunakan untuk membangun sebuah program) secara gratis, yaitu Microsoft Visual Studio. IDE inilah yang akan digunakan untuk membuat aplikasi.

2.6 Visual Studio

Visual studio adalah sebuah *software* yang di referensikan bagi kalangan *developer* windows yang dapat digunakan untuk melakukan pengembangan suatu aplikasi yang dapat berjalan diatas windows. Selain itu, visual studio juga dapat digunakan untuk *developer* aplikasi *windows mobile* yang berjalan di atas .NET *Framework*. Visual Studio mencakup kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa *MSDN Library*). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Pada dasarnya visual studio adalah sebuah bahasa pemrograman komputer. Dimana pengertian dari bahasa pemrograman itu adalah perintah-perintah atau instruksi yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu.

Berikut adalah sejarah perkembangan dari Visual Studio (Strauss, 2020):

Tabel 2.2 Perkembangan Visual Studio

Rilis	Versi	.NET Framework	Tanggal Rilis
Visual Studio 2019	16.0	3.5-4.8	2 April 2019
Visual Studio 2017	15.0	3.5-4.7	7 Maret 2017
Visual Studio 2015	14.0	2.0-4.6	20 Juli 2015
Visual Studio 2013	12.0	2.0-4.5.2	17 Oktober 2013
Visual Studio 2012	11.0	2.0-4.5.2	12 September 2012
Visual Studio 2010	10.0	2.0-4.0	12 April 2010
Visual Studio 2008	9.0	2.0, 3.0, 3.5	19 November 2007
Visual Studio 2005	8.0	2.0, 3.0	7 November 2005
Visual Studio .NET 2003	7.1	1.1	24 April 2003
Visual Studio .NET 2002	7.0	1.0	13 Februari 2002
Visual Studio 6.0	6.0	N/A	Juni 1998
Visual Studio 97	5.0	N/A	Februari 1997

2.7 UML (*Unified Modelling Language*)

Menurut (Rosa & Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, 2016), UML (*Unified Modelling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

Menurut (Mulyani, 2016), UML (*Unified Modeling Language*) adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem.

Dari beberapa penjelasan tersebut dapat disimpulkan bahwa UML (*Unified Modelling Language*) adalah suatu standar bahasa pemodelan visual sebagai sarana perancangan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek.

Menurut (Muslihudin & Oktafianto, 2016), UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisis dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.


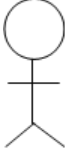

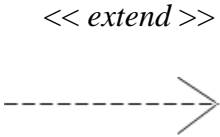
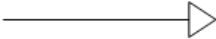
2.7.1 Use Case Diagram

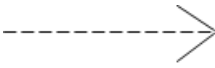
Menurut (Rosa & Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, 2016), *use case* atau *diagram use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat.

Use case diagram merupakan model diagram UML yang digunakan untuk menggambarkan secara ringkas dan menjelaskan proses-proses yang berlangsung di dalam sebuah sistem.

Berikut adalah simbol-simbol yang ada pada *use case diagram* (Rosa & Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, 2016):

Tabel 2.3 Simbol *Use Case Diagram*

Simbol	Nama Simbol	Deskripsi
	<i>Use case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
	<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
	<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
	<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah



		fungsi yang lebih umum dari lainnya.
<pre><<include>></pre> 	<i>Include</i>	Relasi <i>use case</i> dimana proses bersangkutan akan dilanjutkan ke proses yang dituju.



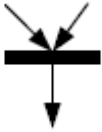
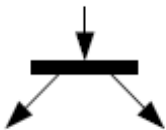

2.7.2 Activity Diagram

Activity diagram merupakan sesuatu yang menjelaskan tentang alir kegiatan dari sebuah sistem atau program yang sedang dirancang, bagaimana proses alir berawal, keputusan yang mungkin terjadi, dan bagaimana sistem akan berakhir. *Activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Berikut adalah simbol-simbol yang ada pada *activity diagram* (Rosa & Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, 2016):

Tabel 2.4 Simbol Activity Diagram

Simbol	Nama Simbol	Deskripsi
	Status awal	Status awal sistem aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.

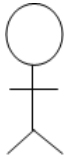

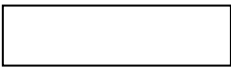

	<i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	<i>Activity Final</i>	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	<i>Join</i>	Digunakan utk menunjukkan kegiatan yang digabungkan.
	<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel.
	<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.


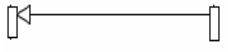
2.7.3 Sequence Diagram

Sequence diagram adalah salah satu diagram UML yang menggambarkan interaksi objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek. Untuk menggambarkan *sequence diagram*, objek-objek yang terlibat harus diketahui dalam sebuah *use case*.

Berikut adalah simbol-simbol yang ada pada *sequence diagram* (Rosa & Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, 2016):

Tabel 2.5 Simbol *Sequence Diagram*

Simbol	Nama Simbol	Deskripsi
	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan dalam menggunakan kata benda diawal frase nama aktor.
	<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
	Objek	Menyatakan objek yang berinteraksi pesan.
	Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
	Pesan tipe <i>send</i>	Menyatakan bahwa suatu objek

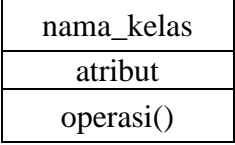






		<p>mengirimkan data/ masukkan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.</p>
	<p>Pesan tipe <i>return</i></p>	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.</p>

2.7.4 Class Diagram

Class diagram adalah diagram UML model statis yang menggambarkan struktur dan deskripsi kelas-kelas serta hubungannya antara kelas yang akan dibuat untuk membangun sistem. Kelas memiliki 3 bagian utama yaitu *name* (identitas pada sebuah kelas), *attribute* (karakteristik data yang dimiliki suatu objek dalam kelas), *operation* (fungsi-fungsi yang dimiliki suatu objek dalam kelas).

Berikut adalah simbol-simbol yang ada pada *class diagram* (Rosa & Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, 2016):

Tabel 2.6 Simbol *Class Diagram*

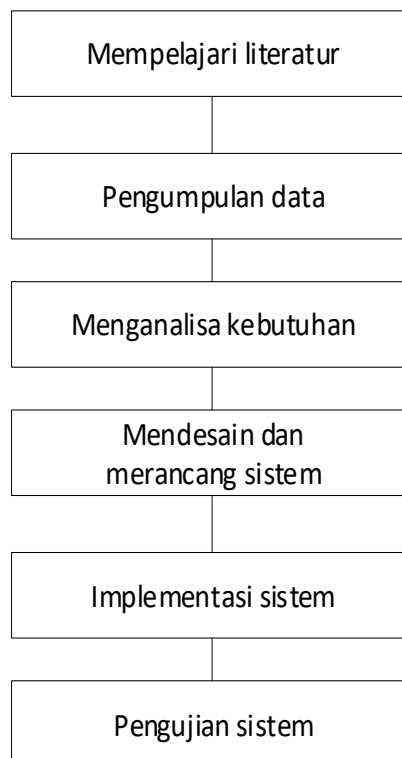
Simbol	Nama Simbol	Deskripsi
	Kelas	Kelas pada struktur sistem.
	Antarmuka/ <i>Interface</i>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
	Asosiasi/ <i>Association</i>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
	Asosiasi berarah/ <i>Directed Association</i>	Relasi antarkelas dengan makna kelas satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
	Generalisasi	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
	Kebergantungan/ <i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antarkelas.
	Agregasi/ <i>Aggregation</i>	Relas antarkelas dengan makna semuabagian (<i>whole-part</i>).

BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Untuk membantu dalam penyusunan penelitian ini, maka perlu adanya susunan kerangka kerja yang jelas tahapan-tahapannya. Kerangka kerja ini merupakan langkah-langkah yang akan dilakukan dalam penyelesaian masalah yang akan dibahas. Adapun kerangka kerja penelitian yang di gunakan seperti terlihat pada gambar 3.1.



Gambar 3.1 Kerangka Penelitian

Berdasarkan kerangka kerja penelitian yang telah digambarkan di atas, maka dapat diuraikan pembahasan masing-masing tahap dalam penelitian adalah sebagai berikut :

1. Mempelajari Literatur

Pada tahap ini dilakukan pencarian landasan-landasan teori yang diperoleh dari berbagai buku dan juga internet untuk melengkapi perbendaharaan konsep dan teori, sehingga memiliki landasan dan keilmuan yang baik dan sesuai.

2. Pengumpulan Data

Metode pengumpulan data yang dilakukan dalam penelitian ini adalah:

a. Analisa Masalah

Analisa masalah dilakukan dengan pengamatan pada masyarakat sekitar tentang kedisiplinan tepat waktu dalam melakukan aktivitas atau kegiatannya sehari-hari.

b. Studi Pustaka

Pengumpulan data dengan menggunakan atau mengumpulkan sumber-sumber tertulis, dengan cara membaca, mempelajari dan mencatat hal-hal penting yang bersumber dari buku, jurnal dan internet yang berhubungan dengan masalah yang sedang dibahas guna memperoleh gambaran secara teoritis.

3. Menganalisa Kebutuhan

Pada tahap ini dilakukan identifikasi masalah pada sistem yang sedang berjalan. Dengan demikian, diharapkan peneliti dapat menemukan permasalahan

yang terjadi pada prosesnya sehingga peneliti dapat mencari solusi dari permasalahan tersebut.

4. Mendesain dan Merancang Sistem

Pada Tahap ini dilakukan mendesain sistem dengan menggunakan permodelan UML (*Unified Modelling Language*) dan pada tahap ini dilakukan perancangan antarmuka aplikasi yang akan dibuat.

5. Implementasi Sistem

Implementasi sistem dilakukan sesuai desain dan rancangan antarmuka aplikasi yang akan dibangun. Pada tahap ini melakukan pengkodean atau pembuatan program aplikasi yang dirancang dapat digunakan oleh pengguna.

6. Pengujian Sistem

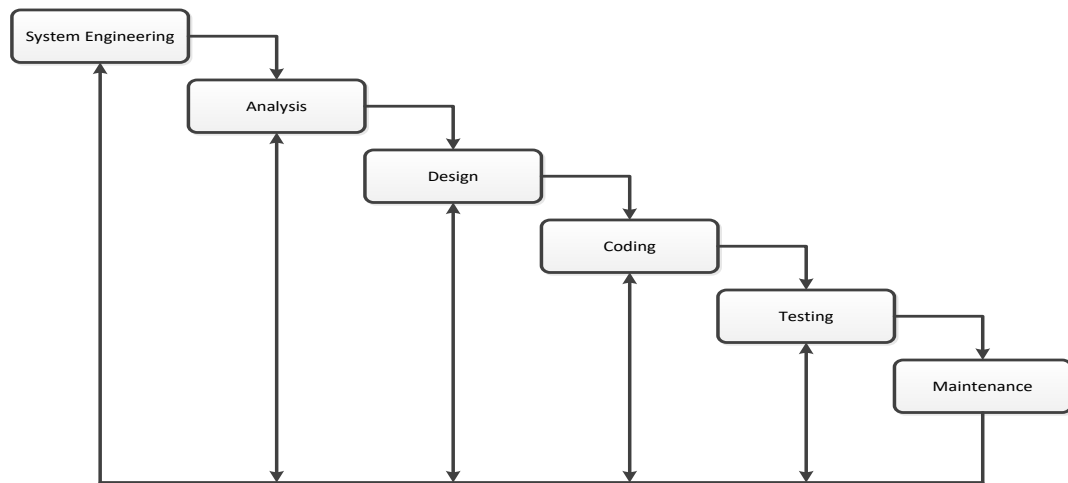
Pada tahap ini aplikasi yang sudah dirancang sebelumnya telah selesai dan dilakukan tahapan pengujian aplikasi apakah ada *error* atau kerusakan pada aplikasi yang telah dirancang.

3.2 Metodologi Penelitian

3.2.1. Metode Pengembangan Perangkat Lunak

Metodologi yang digunakan untuk membangun sistem ini adalah model *waterfall*. Model ini merupakan sebuah pendekatan terhadap pengembangan perangkat lunak yang sistematis, dengan beberapa tahapan, yaitu: *System Engineering, Analysis, Design, Coding, Testing* dan *Maintenance*.

Adapun tahapan-tahapan dari paradigma *waterfall* sebagai berikut:



Gambar 3.2 Paradigma Waterfall (Classic Life Cycle)

(Sumber: Iqbal, 2017)

Berikut penjelasan metodologi *Waterfall*:

1. *System Engineering* adalah Permodelan ini diawali dengan mencari kebutuhan dari keseluruhan sistem yang akan diaplikasikan ke dalam bentuk *software*. Hal ini sangat penting, mengingat *software* harus dapat berinteraksi dengan elemen-elemen yang lain seperti *hardware*, *database*, dan sebagainya.
2. Analisis adalah tahap menganalisa hal-hal yang diperlukan dalam pelaksanaan proyek pembuatan *software*. Dalam hal ini dilakukan dengan menganalisa dokumen-dokumen yang digunakan dalam informasi produktivitas diri.
3. *Design* adalah tahap penterjemah dari keperluan-keperluan yang dianalisis dalam bentuk yang lebih mudah dimengerti oleh pemakai.
4. *Coding* adalah tahap penterjemah data/ pemecahan masalah *software* yang telah dirancang dalam bahasa pemograman yang telah ditentukan dan

digunakan dalam pembuatan sistem menggunakan *software Visual Studio 2017* dengan bahasa pemrograman C#.

5. *Testing* adalah tahap pengujian terhadap program yang telah dibuat. Pengujian ini dimulai dengan membuat suatu uji kasus untuk setiap fungsi pada perangkat lunak, kemudian dilanjutkan dengan pengujian terhadap modul-modul dan terakhir pada tampilan antar muka untuk memastikan tidak ada kesalahan dan semua berjalan dengan baik dan input yang diberikan hasilnya sesuai dengan yang diinginkan.
6. *Implementation* adalah menguji *software* aplikasi yang telah dibuat dan dirancang dengan membuat aplikasi *software* menjadi apk, dan diimplementasikan pada perangkat pendukung.
7. *Maintenance* adalah perangkat lunak yang telah dibuat dapat mengalami perubahan sesuai permintaan pemakai. Pemeliharaan dapat dilakukan jika ada permintaan tambahan fungsi sesuai dengan keinginan pemakai ataupun adanya pertumbuhan dan perkembangan baik perangkat lunak maupun perangkat keras.

3.3 Analisis Sistem

Analisis sistem merupakan bagian yang sangat penting, karena apabila terjadi kesalahan dalam tahap ini, maka akan mengakibatkan kesalahan pada tahap selanjutnya. Pada bagian analisis sistem ini akan dibahas tentang analisis masalah, analisis sistem yang sedang berjalan, analisis sistem yang dikembangkan, analisis sumber pengetahuan, analisis kriteria, hasil dan analisis kebutuhan.

Analisis masalah adalah penguraian dari suatu masalah yang utuh ke dalam bagian-bagian komponennya dengan maksud mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan.

Berdasarkan hasil dari penelitian penulis dengan mengumpulkan data dari masyarakat sekitar tentang aktivitas harian yang dilakukan. Masih terdapatnya ketidaktepatan waktu dalam menjalani aktivitas yang akan dijalani tersebut. Oleh karena itu berdasarkan analisis masalah yang terjadi, maka melalui sistem yang akan dirancang penulis ini diharapkan menjadi pilihan alternatif dalam memberikan pengingat pada jadwal aktivitas yang akan dijalani oleh masyarakat tersebut dengan mudah. Sehingga dapat membuat masyarakat menjadi lebih produktif dalam menjalani aktivitas.

3.4 Analisis Kebutuhan

3.4.1. Analisis Kebutuhan Fungsional

Terdapat tiga alat penelitian dalam penyelesaian aplikasi *self productivity list* ini, yaitu :

1. *Smartphone* Android

Smartphone Android yang digunakan untuk menguji coba aplikasi ini, memiliki spesifikasi sebagai berikut:

- 1) CPU : *Qualcomm MSM8909 1,6 GHz*
- 2) *Memory Internal* : *1 GB RAM, 8 GB ROM*
- 3) *Memory External* : *8 GB*

- 4) *Operating System* : *Android OS, V5.0 (Lollipop)*
- 5) *Tipe layar* : *Corning Gorilla Glass 3 Multi Touch Screen*
- 6) *Ukuran Layar* : *720 x 1280 pixel*

2. *Hardware*

Hardware yang digunakan memiliki spesifikasi sebagai berikut :

- 1) *Processor* : *Intel® Core™ i3*
- 2) *Memory* : *2 GB DDR 3*
- 3) *Harddisk* : *500 GB*
- 4) *Operating System* : *Windows 10 64bit*
- 5) *Display* : *14 Inch WXGA (1366 x 768)*
- 6) *Ukuran Layar* : *1366 x 768 pixel*

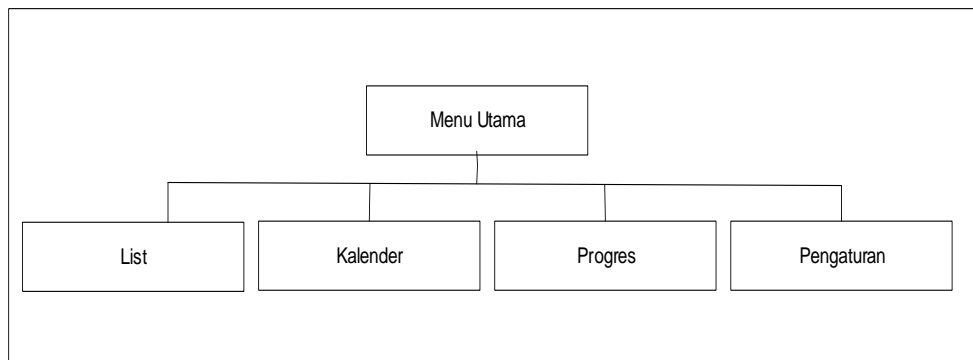
3. *Software*

Software yang digunakan dalam pembuatan aplikasi *self productivity list* adalah sebagai berikut:

- 1) *Visual Studio .Net 2019*
- 2) *SQLite DBrowser*
- 3) *Java SDK 8.201*

3.5 Perancangan Arsitektur Navigasi

Dari aplikasi *self productivity list* ini, tampilan awalnya adalah tampilan Form Utama yang didalamnya terdapat menu lain dan keseluruhan dari tampilan yang ada pada aplikasi ini, dapat dilihat pada gambar dibawah ini :

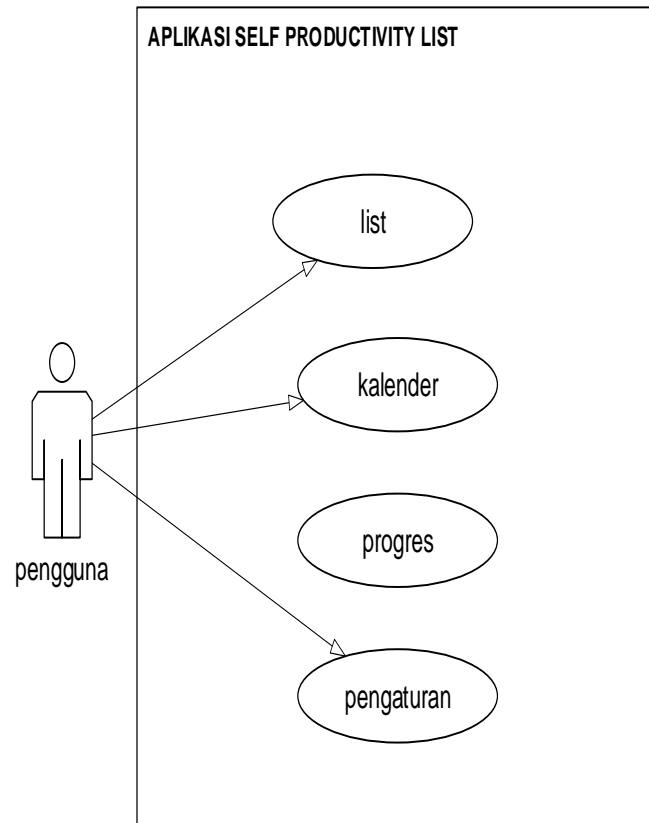


Gambar 3.3 Struktur Arsitektur Navigasi

3.6 Perancangan Sistem

3.6.1. Use Case Diagram

Untuk mendapatkan informasi dari sebuah sistem yang dibuat, maka penulis menggunakan *use case diagram*. Dengan diagram ini, proses yang terjadi pada sebuah aplikasi akan dapat diketahui. *Use case diagram* dari aplikasi *self productivity list* dapat dilihat dibawah ini :



Gambar 3.4 Use Case Diagram Self Productivity List

1. Definisi Aktor

Berikut adalah deskripsi pendefinisian aktor pada aplikasi *self productivity list*:

Tabel 3.1 Definisi Aktor

Aktor	Deskripsi
Pengguna	Orang yang menggunakan aplikasi <i>self productivity list</i> .

2. Definisi *Use case*

Berikut adalah deskripsi pendefinisian *Use case* pada aplikasi *self productivity list*:

Tabel 3.2 Definisi *Use case*

No	<i>Use case</i>	Deskripsi
1.	<i>List</i>	Merupakan menu yang berfungsi untuk membuat dan melihat jadwal aktivitas harian.
2.	Kalender	Merupakan menu yang berfungsi untuk melihat jadwal harian berdasarkan tanggal dari kalender.
3.	Progres	Merupakan menu yang berfungsi untuk melihat progres dari aktivitas harian yang telah dilakukan.
4.	Pengaturan	Merupakan menu yang berfungsi untuk mengubah pengaturan pada tampilan <i>list</i> jadwal.

3. Skenario *Use case*

Berikut adalah skenario jalannya masing-masing *use case* yang telah didefinisikan sebelumnya :

a. Skenario *Use case List*

Nama *Use case* : *List*

Skenario :

Tabel 3.3 Skenario *Use case List*

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu <i>List</i>	
	2. Menampilkan tampilan jadwal aktivitas harian dan membuat jadwal baru.

b. Skenario *Use case* KalenderNama *Use case* : Kalender

Skenario :

Tabel 3.4 Skenario *Use case* Kalender

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu Kalender	
	2. Menampilkan form kalender yang akan menampilkan jadwal aktivitas yang pernah dibuat.

c. Skenario *Use case* ProgresNama *Use case* : Progres

Skenario :

Tabel 3.5 Skenario *Use case* Progres

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu Progres	
	2. Menampilkan form progres aktivitas yang sudah dilakukan sebelumnya.

d. Skenario *Use case* PengaturanNama *Use case* : Pengaturan

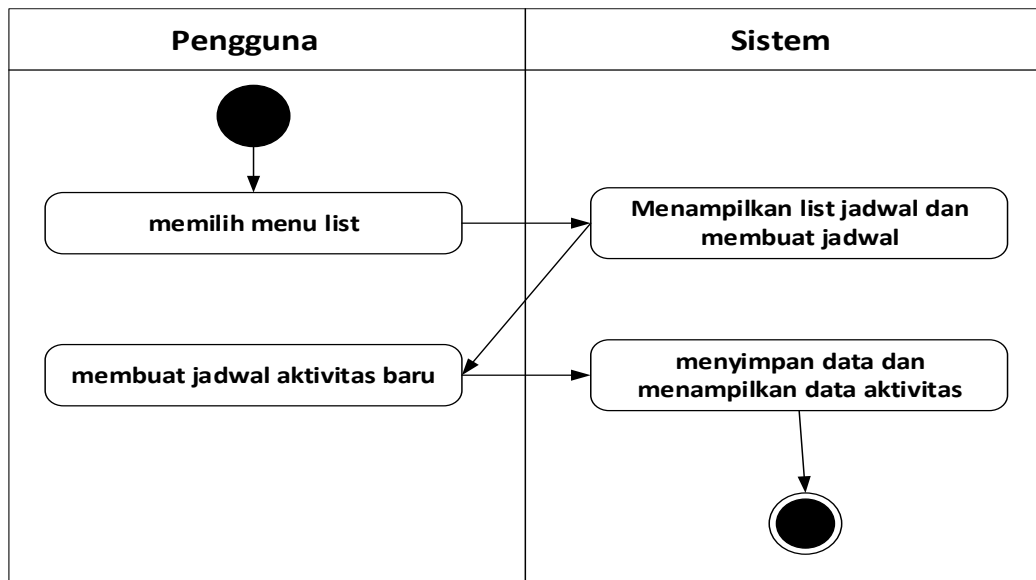
Skenario :

Tabel 3.6 Skenario *Use case* Pengaturan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih menu Pengaturan	
	2. Menampilkan form pengaturan yang dapat mengubah <i>list</i> tampilan aktivitas harian.

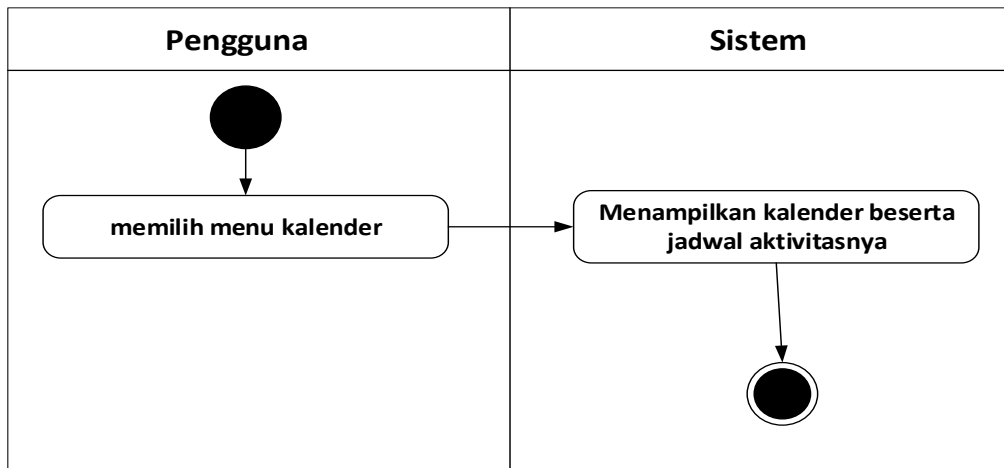
3.6.2. Activity Diagram

1. Activity Diagram List



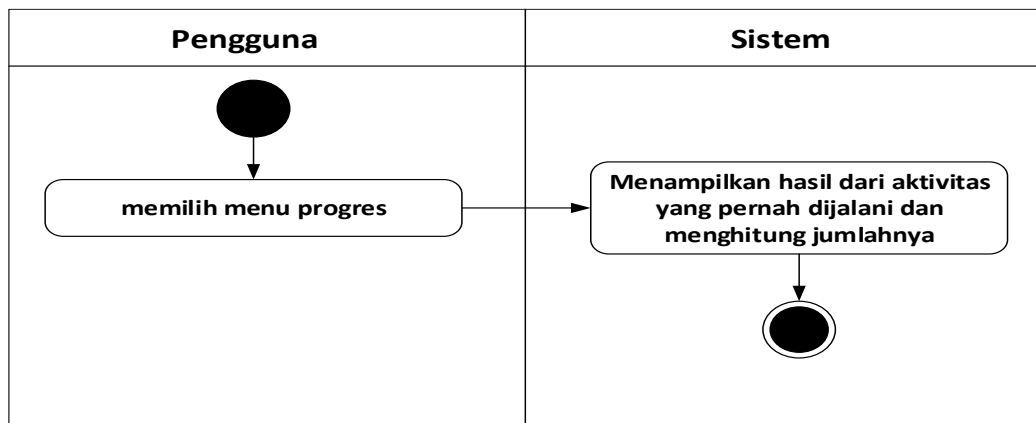
Gambar 3.5 Activity Diagram List

2. Activity Diagram Kalender



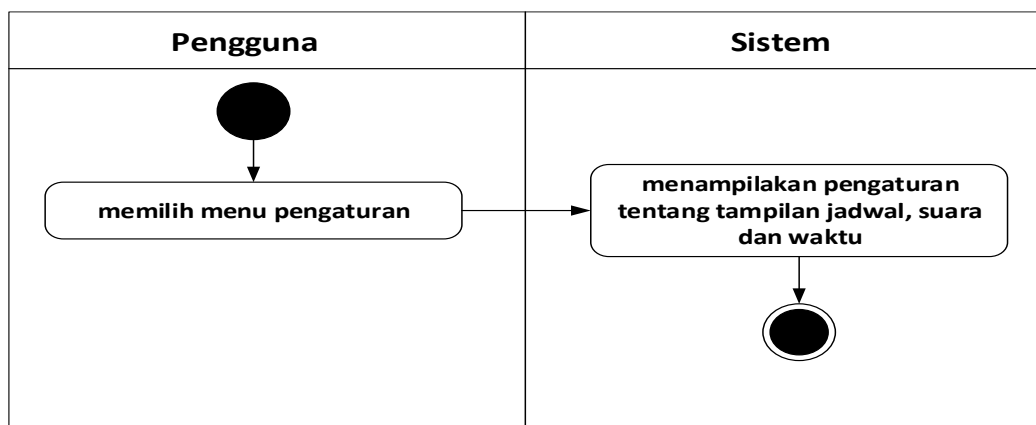
Gambar 3.6 Activity Diagram Kalender

3. Activity Diagram Progres



Gambar 3.7 Activity Diagram Progres

4. Activity Diagram Pengaturan

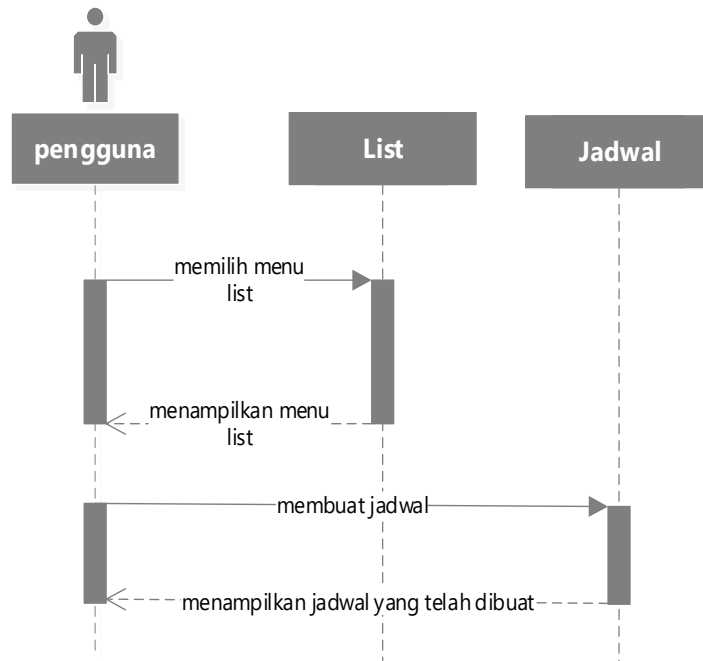


Gambar 3.8 Activity Diagram Pengaturan

3.6.3. Sequence Diagram

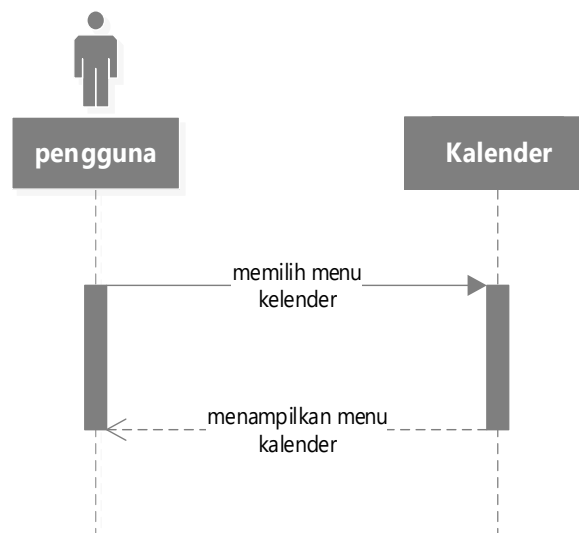
Berikut adalah *sequence diagram* aplikasi *self productivity list*:

1. Diagram Sequence List



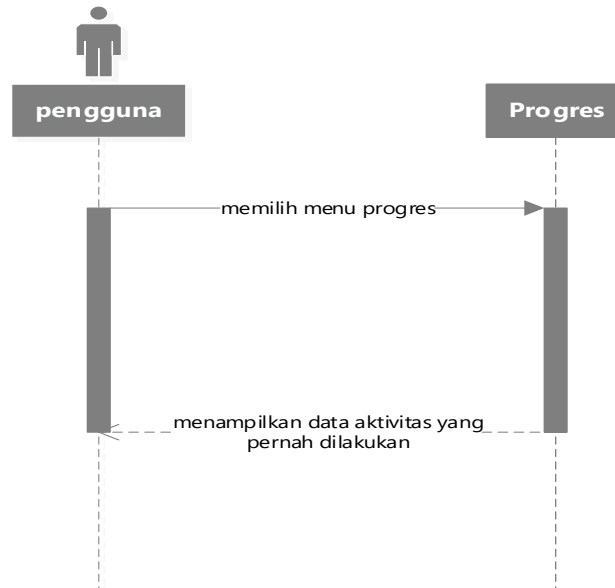
Gambar 3.9 *Diagram Sequence List*

2. *Diagram Sequence Kalender*



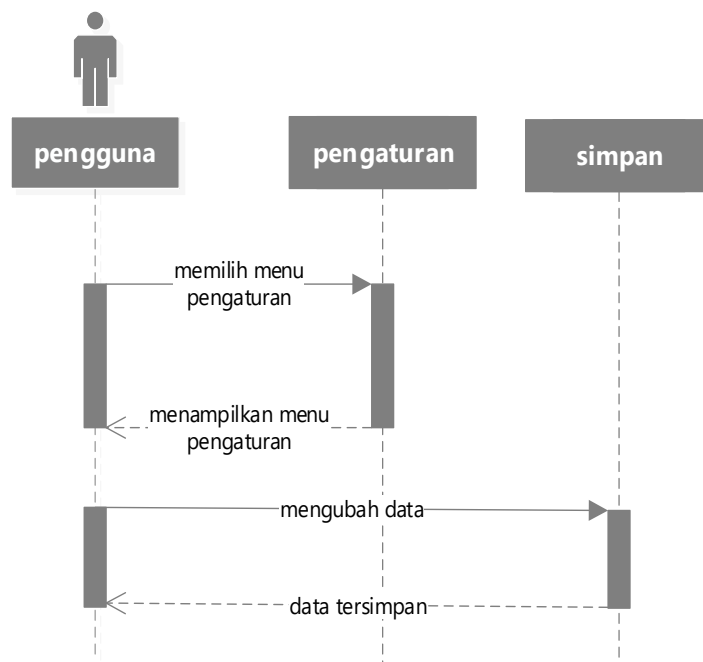
Gambar 3.10 *Diagram Sequence Kalender*

3. *Diagram Sequence Progres*



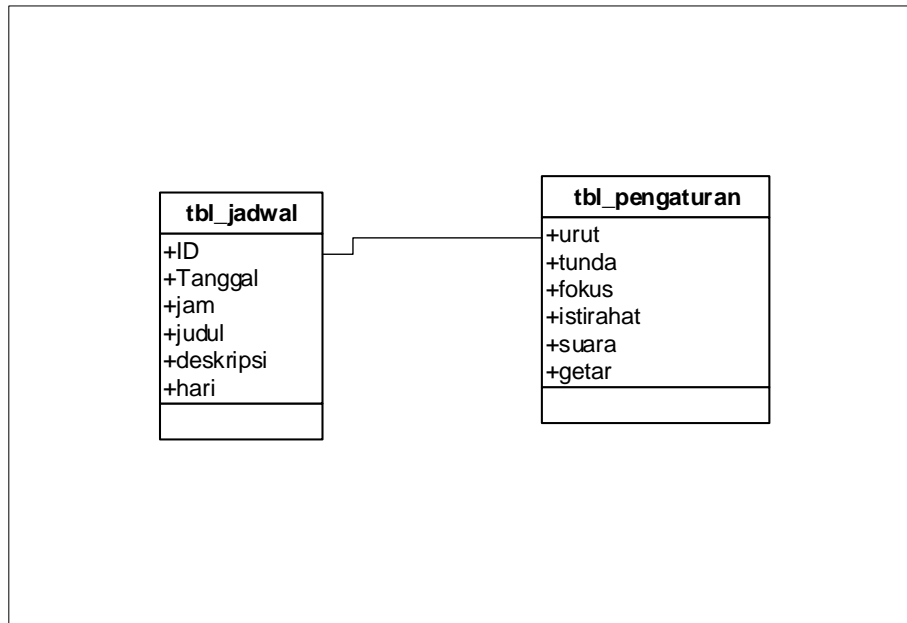
Gambar 3.11 *Diagram Sequence Progres*

4. *Diagram Sequence Pengaturan*



Gambar 3.12 *Diagram Sequence Pengaturan*

3.6.4. Class Diagram



Gambar 3.13 Class Diagram Self Productivity List

3.6.5. Perancangan Tabel

Pada tahap perancangan struktur file untuk mempermudah dalam mengetahui suatu nilai atau tipe data yang ada pada file penyimpanan ini akan dijelaskan mengenai perancangan basis data yang akan digunakan. Penyusunan tabel ini pada dasarnya digunakan untuk memudahkan dalam pemasukan dengan penyimpanan data yang sesuai dengan kelompok dari data atau informasi tersebut. Tabel-tabel yang ada di bawah ini tersimpan dalam suatu *database* yang bernama **db_jadwal**.

1. Tabel Jadwal

Tabel 3.7 Tabel Jadwal

Field	Type	Keterangan
ID	Integer	<i>Primary key</i>
tanggal	Text	
Jam	Text	
Judul	Text	
Deskripsi	Text	
Hari	Text	

2. Tabel Pengaturan

Tabel 3.8 Tabel Pengaturan

Field	Type	Keterangan
Urut	Text	
Tunda	Text	
Fokus	Text	
Istirahat	Text	
Suara	Text	
Getar	Text	

3.6.6. Perancangan Antarmuka (*User Interface*)

Perancangan Antarmuka adalah rancangan yang dilakukan untuk memberikan gambaran aplikasi yang akan ditampilkan secara sederhana kepada

pengguna. Diharapkan pengguna yang menggunakan aplikasi ini dapat dengan mudah mengerti fungsi dari tombol yang ada pada aplikasi. Dalam aplikasi *self productivity list* ini, terdapat beberapa bagian tampilan yang memiliki fungsi berbeda pada setiap tombolnya. Fungsi-fungsi dari tombol yang ada pada setiap bagian tampilan akan dijelaskan dan dapat dilihat pada gambar berikut :

1. Rancangan Tampilan Form Utama

Rancangan Tampilan Form Utama adalah tampilan yang pertama kali ditampilkan. Tampilan ini disebut dengan tampilan Form utama, yang akan menampilkan *screen opening*.

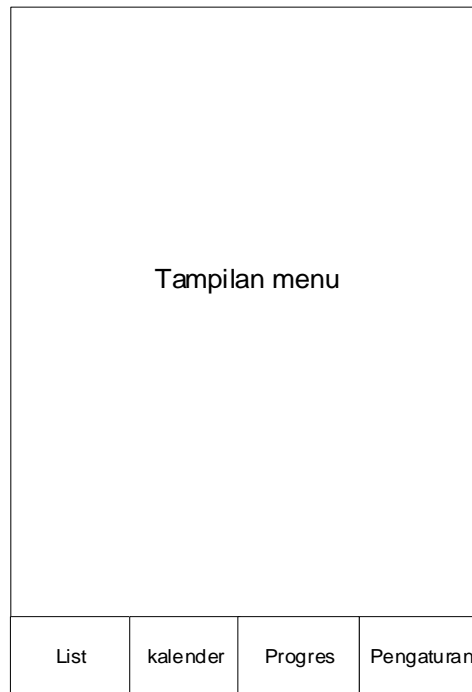


Gambar 3.14 Rancangan Tampilan Awal

2. Rancangan Tampilan Menu Utama

Rancangan tampilan menu utama adalah tampilan yang dapat digunakan oleh pengguna untuk menuju ke tampilan yang diinginkan dengan memilih menu yang ada pada tampilan.

Berikut fungsi dari 4 tombol yang ada pada menu utama :



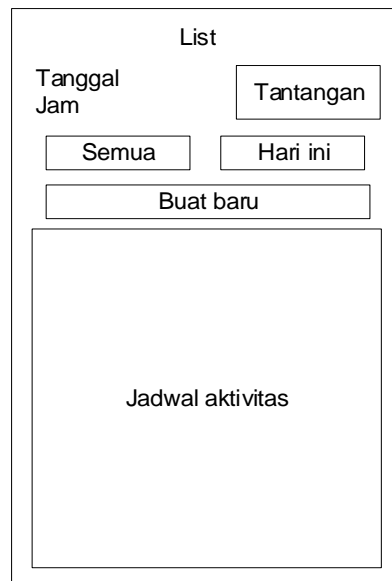
Gambar 3.15 Rancangan Tampilan Menu Utama

Berikut fungsi dari 4 tombol yang ada pada menu utama :

- 1) Tombol list berfungsi untuk melihat jadwal aktivitas harian dan membuat jadwal aktivitas berikutnya.
- 2) Tombol kalender berfungsi untuk melihat jadwal aktivitas harian berdasarkan tanggal.
- 3) Tombol progres berfungsi untuk melihat aktivitas jadwal yang telah dilakukan.
- 4) Tombol pengaturan berfungsi untuk mengubah data pengaturan pada aplikasi.

3. Rancangan Tampilan List

Rancangan Tampilan List ini berfungsi untuk, melihat jadwal aktivitas harian dan membuat jadwal aktivitas berikutnya.



Gambar 3.16 Rancangan Tampilan *List*

4. Rancangan Tampilan Kalender

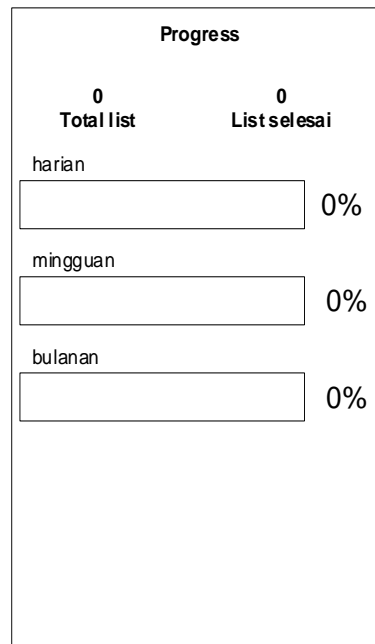
Rancangan ini adalah rancangan yang menampilkan aktivitas harian berdasarkan tanggal dari kalender.



Gambar 3.17 Rancangan Tampilan Kalender

5. Rancangan Tampilan Menu Progres

Rancangan ini adalah rancangan yang menampilkan jadwal aktivitas yang pernah dilakukan.



Gambar 3.18 Rancangan Tampilan Menu Progres

6. Rancangan Tampilan Menu Pengaturan

Rancangan ini adalah rancangan yang berfungsi untuk mengubah pengaturan pada aplikasi.

Pengaturan	
Urutkan list berdasarkan	<input type="text"/>
Waktu tunda	<input type="text"/>
Pengaturan waktu fokus	
Durasi	<input type="text"/>
Istirahat	<input type="text"/>
Notifikasi	
Suara	<input type="checkbox"/>
Getar	<input type="checkbox"/>
<input type="button" value="Hubungi kami"/>	
<input type="button" value="Simpan pengaturan"/>	

Gambar 3.19 Rancangan Tampilan Menu Pengaturan

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem Yang Digunakan

Dalam perancangan aplikasi *self productivity list* ini, penulis menggunakan program aplikasi berbasis android dengan C# sebagai bahasa pemrogramannya. Aplikasi yang dibuat cukup mudah dalam pengoperasiannya untuk pengguna. Pengoperasiannya hanya perlu melakukan klik pada tombol yang telah disediakan untuk mengakses menu lainnya. Pengguna yang ingin membuat *list* jadwal maupun melihat jadwal yang telah dibuat sebelumnya hanya tinggal memilih menu yang berada dibawah.

Tahapan implementasi yang dilakukan untuk menyelesaikan perancangan aplikasi *self productivity list* ini diperlukan informasi mengenai penyediaan perangkat keras (*hardware*) dan perangkat lunak (*software*).

Berikut disediakan perangkat keras dan perangkat lunak yang dibutuhkan.

1. Spesifikasi Perangkat Keras

Aplikasi *self productivity list* ini telah diuji pada *smartphone* dengan spesifikasi perangkat keras sebagai berikut:

- 1) *CPU* : *Qualcomm MSM8909 1,2 GHz, GPU*
- 2) *Memory Internal* : *1 GB RAM, 8 GB ROM*
- 3) *Memory External* : *8 GB*

- 4) *Operating System* : *Android OS, 4.0 (Kitkat)*
- 5) *Tipe Layar* : *Corning Gorilla Glass 3*
- 6) *Ukuran Layar* : *720 x 1280 pixel*

2. Spesifikasi Perangkat Lunak

Aplikasi ini dijalankan pada perangkat lunak dengan spesifikasi sebagai berikut:

- 1) *Sistem Operasi : Android OS, 5.0 (Lollipop)*

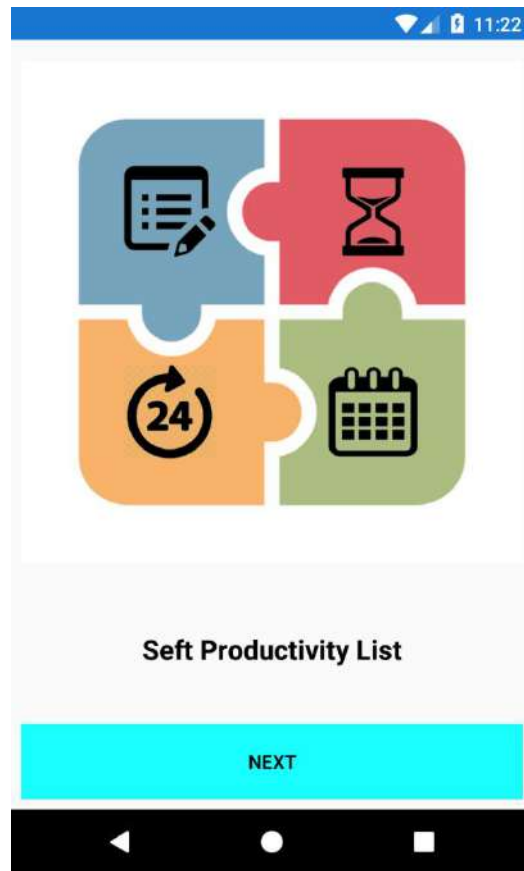
4.2 Implementasi *Interface*

Implementasi *interface* dilakukan untuk mengetahui apakah sistem yang telah dibuat benar-benar sesuai dengan yang telah direncanakan. Pada tahap ini akan dijelaskan bagaimana program sistem ini bekerja, dengan memberikan tampilan sistem atau aplikasi yang dibangun.

Adapun tampilan dari menu-menu aplikasi *self productivity list* adalah sebagai berikut :

1. Tampilan Awal

Tampilan awal merupakan tampilan yang pertama kali tampil ketika pengguna baru menginstal aplikasi ini.



Gambar 4.1 Tampilan Awal

2. Tampilan Menu Utama

Tampilan menu utama terdiri dari menu *list*, kalender, progres, dan pengaturan yang berfungsi untuk menghubungkan ke menu yang lainnya.



Gambar 4.2 Tampilan Menu Utama

3. Tampilan Tantangan

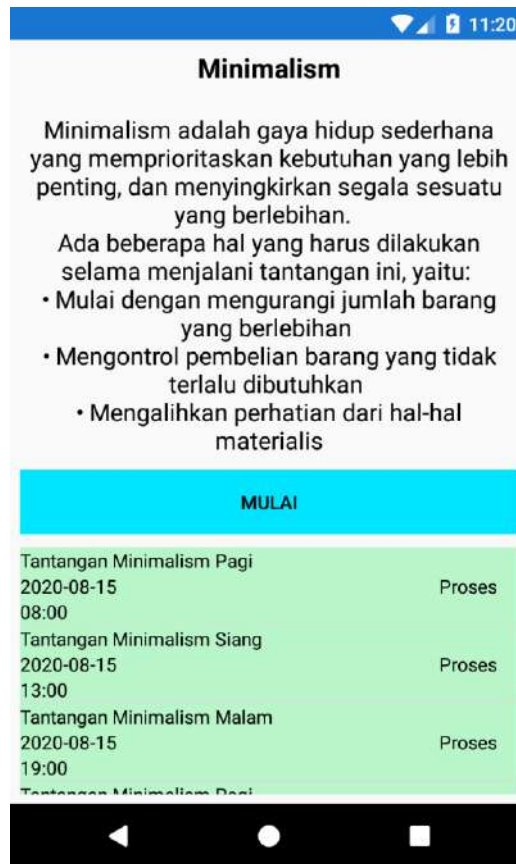
Tampilan tantangan merupakan tampilan yang berisi kumpulan aktivitas yang akan dilakukan pengguna selama 30 hari berturut-turut.



Gambar 4.3 Tampilan Tantangan

4. Tampilan Detail Tantangan

Tampilan detail tantangan merupakan tampilan yang berisi informasi dari kegiatan tantangan yang akan dilakukan oleh pengguna. Dengan melakukan klik mulai, maka jadwal kegiatan akan dibuat secara otomatis dan akan berlaku mulai ke esokan harinya.



Gambar 4.4 Tampilan Detail Tantangan

5. Tampilan Jadwal Rancangan

Tampilan jadwal rancangan merupakan menu untuk membuat jadwal yang akan dilakukan oleh pengguna dengan membuat judul dan waktunya secara manual.



Gambar 4.5 Tampilan Jadwal Rancang

6. Tampilan Jadwal Template

Tampilan jadwal template merupakan menu untuk membuat jadwal kegiatan tambahan yang akan dilakukan oleh pengguna. Pada menu jadwal template ini pengguna hanya tinggal memilih jadwal yang akan dilakukan tanpa harus mengetik jadwal secara manual.



Gambar 4.6 Tampilan Jadwal Template

7. Tampilan Kalender

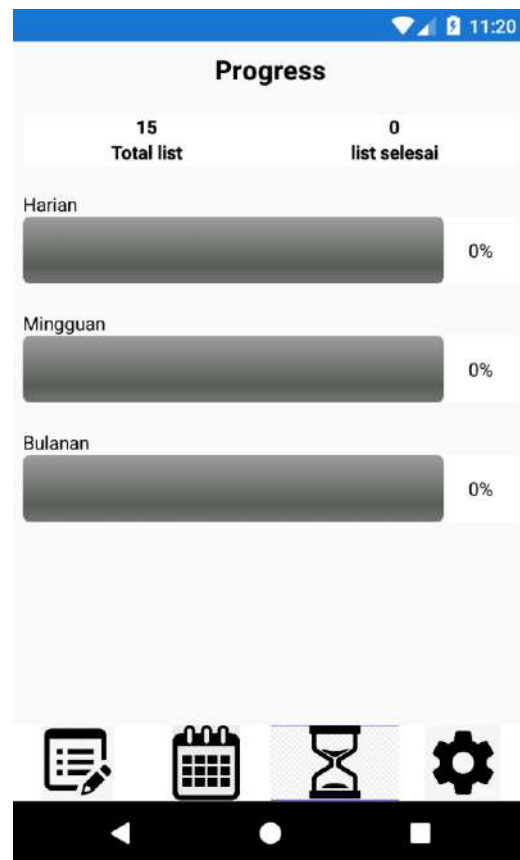
Tampilan kalender berfungsi untuk melihat jadwal pada bulan tersebut, yang akan dilakukan oleh pengguna dengan melakukan klik pada tanggal yang tersedia.



Gambar 4.7 Tampilan Kalender

8. Tampilan Progres

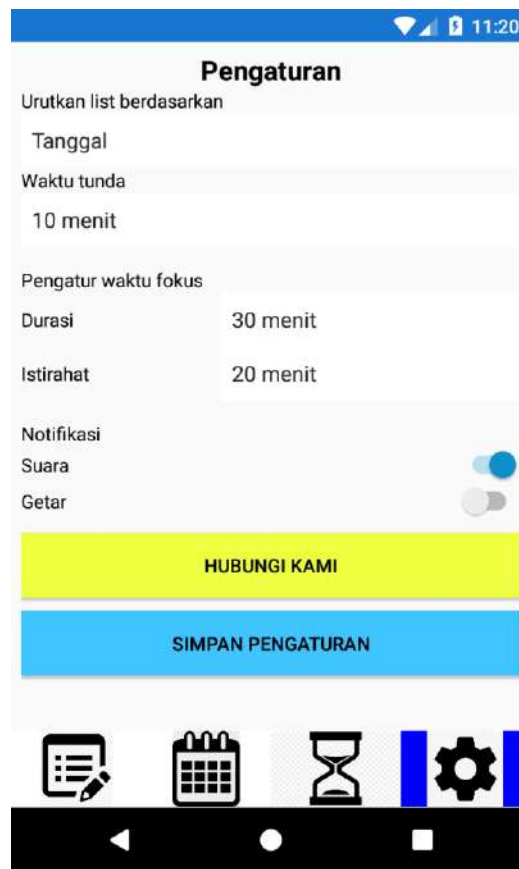
Tampilan progres merupakan tampilan jadwal kegiatan yang telah dilakukan oleh pengguna. Didalam menu ini pengguna dapat melihat persentase dari seluruh jadwal yang ada dan pengguna dapat melihat jadwal yang telah dilakukan maupun yang belum dilakukan



Gambar 4.8 Tampilan Progres

9. Tampilan Pengaturan

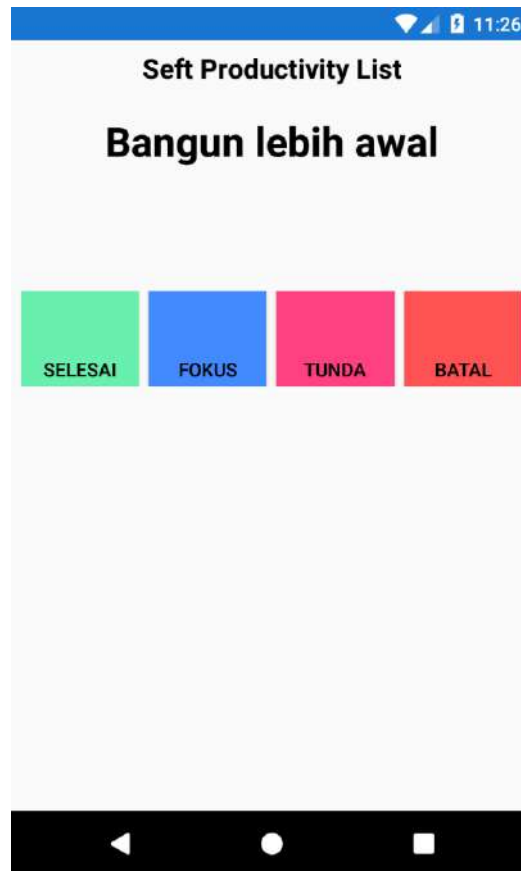
Tampilan pengaturan berfungsi untuk mengatur aplikasi seperti mengurutkan *list*, waktu tunda jadwal dan lain-lain. Di menu ini juga pengguna dapat mengirimkan email berupa keluhan ataupun masukan mengenai aplikasi tersebut.



Gambar 4.9 Tampilan Pengaturan

10. Tampilan Notifikasi

Tampilan notifikasi merupakan tampilan yang akan tampil ketika ada notifikasi dari aplikasi untuk melakukan kegiatan sesuai dengan jadwal yang telah ditentukan oleh pengguna sebelumnya.



Gambar 4.10 Tampilan Notifikasi

4.3 Pengujian Aplikasi *Self Productivity List*

Pengujian aplikasi *self productivity list* ini digunakan untuk menguji sistem pada salah satu menu dimana data yang digunakan adalah proses membuat jadwal kegiatan menggunakan template yang telah disediakan. Cara menggunakan aplikasi *self productivity list* adalah sebagai berikut :

1. Langkah awalnya pengguna menjalankan aplikasi dan akan ditampilkan menu utama aplikasi.



Gambar 4.11 Menu Utama

2. Kemudian pengguna melakukan klik pada menu buat baru, lalu aplikasi akan menampilkan pilihan berupa rancang dan template. Disini pengguna memilih menu template yang tersedia.



Gambar 4.12 Pilih Menu

3. Setelah melakukan klik pada tombol template, aplikasi akan menampilkan tampilan baru yaitu tampilan template yang dapat dipilih oleh pengguna.



Gambar 4.13 Tampilan Template

4. Didalam menu ini pengguna hanya tinggal memilih jadwal sesuai keinginan yang akan dilakukan. Tampilan ini terdapat 3 kategori yaitu rutinitas, spiritual, dan kesehatan.
5. Setelah memilih jadwal kegiatan yang akan dilakukan di menu ini, aplikasi akan menampilkan detail dari jadwal template yang pengguna hanya tinggal menentukan tanggal dan jam untuk melakukan kegiatan tersebut.



Gambar 4.14 Hasil Identifikasi

6. Kemudian setelah pengguna selesai mengisi data yang diperlukan tersebut, pengguna dapat melakukan klik pada tombol simpan untuk menyimpan jadwal sesuai dengan tanggal dan jam yang telah diatur sebelumnya.

4.4 Pengujian Kuesioner

Pengujian kuesioner dilakukan dengan turun langsung ke lapangan yaitu masyarakat luas untuk mengetahui pendapat responden terhadap aplikasi *Self Productivity List*.

Pengujian ini dilakukan oleh 10 orang pengguna untuk menguji kinerja dari aplikasi *Self Productivity List*.

Tabel 4.1 Pengujian Kuesioner

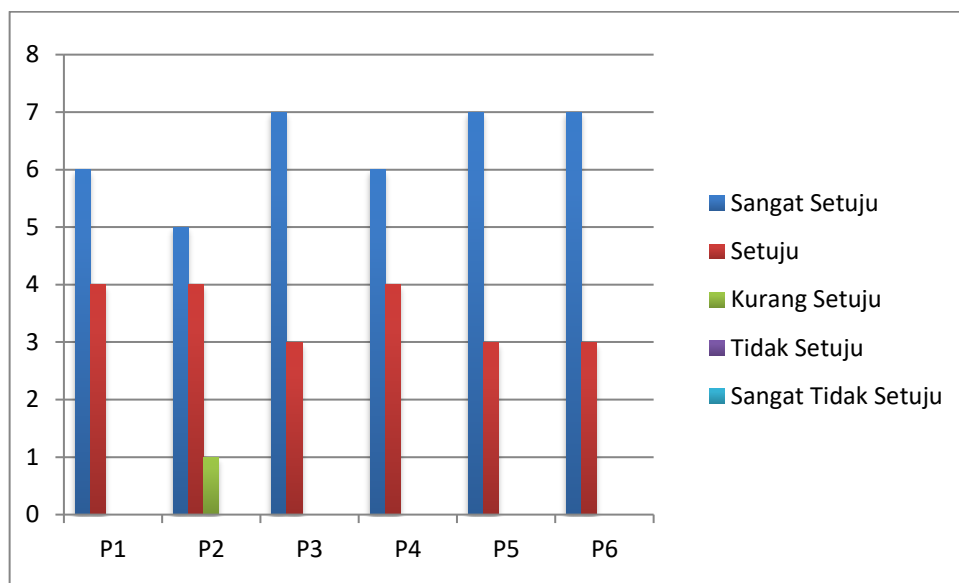
No	Pertanyaan	Penilaian				
		SS	S	KS	TS	STS
1	Aplikasi <i>self productivity list</i> mudah digunakan	6	4	0	0	0
2	Tampilan aplikasi <i>self productivity list</i> menarik	5	4	1	0	0
3	Aplikasi <i>self productivity list</i> bermanfaat bagi pengguna	7	3	0	0	0
4	Informasi pada aplikasi <i>self productivity list</i> mudah dipahami	6	4	0	0	0
5	Aplikasi <i>self productivity list</i> memudahkan pengguna untuk mengatur aktivitas dan catatan-catatan penting	7	3	0	0	0
6	Respon aplikasi <i>self productivity list</i> terhadap pengguna cepat	7	3	0	0	0
Total Skor		38	21	1	0	0
Persentase		63%	35%	1%	0%	0%

Keterangan: SS= Sangat Setuju, S= Setuju, KS= Kurang Setuju, TS= Tidak Setuju, STS= Sangat Tidak Setuju.

$$P = (\text{Skor} / \text{Smax}) \times 100\%$$

Tabel 4.1 merupakan data hasil dari pengujian kuesioner yang telah diambil dari 10 orang pengguna aplikasi yang terdiri dari 6 buah pertanyaan mengenai aplikasi *self productivity list*.

Berikut adalah hasil pengujian kuesioner berdasarkan grafik.



Gambar 4.15 Grafik Kuesioner

Berdasarkan hasil pengujian kuesioner yang diperoleh maka dapat disimpulkan bahwa aplikasi *self productivity list* sangat bermanfaat dan memudahkan pengguna untuk mengatur aktivitas dan catatan-catatan penting. Respon aplikasi *self productivity list* terhadap pengguna juga sangat cepat. Hal ini dilihat dari tingkat kepuasan pengguna yang memberikan respon sangat setuju dengan menghasilkan persentase interpretasi 63%.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pembuatan aplikasi *self productivity list*, maka dapat diambil kesimpulan sebagai berikut:

1. Untuk dapat memudahkan pengguna dalam melakukan aktivitas harian tersebut, sistem yang telah dirancang pada aplikasi *self productivity list* ini dapat membantu mengelola aktivitas yang akan dilakukan pengguna berdasarkan jadwal pada tanggal yang telah ditetapkan sebelumnya secara lebih mudah dengan melakukan klik pada aplikasi.
2. Fitur Tantangan akan sangat membantu *user* untuk menciptakan suatu kebiasaan baru hingga kebiasaan tersebut akan menjadi otomatis (terus menerus dilakukan).
3. Aplikasi yang dibangun untuk dapat mengelola aktivitas harian pengguna dibuat dengan sistem berbasis android untuk dapat digunakan pada *smartphone* dengan sistem operasi android yang telah banyak beredar dan digunakan oleh banyak pengguna.

5.2 Saran

Berikut adalah saran dari penulis agar aplikasi *self productivity list* dapat bermanfaat dan dikembangkan menjadi lebih baik lagi :

1. Pada sistem yang telah dibangun ini, pengguna dapat membuat *list* aktivitas sendiri maupun dengan menggunakan template yang telah disiapkan dan data yang tersimpan masih secara *offline*. Untuk pengembangan selanjutnya, perlu dibuat sistem yang dapat menyimpan data aktivitas pengguna secara *online* sehingga pengguna dapat memulihkan data apabila *smartphone* tersebut terinstall ulang.
2. Aplikasi *self productivity list* ini hanya dapat digunakan pada *smartphone* dengan sistem operasi android minimum versi 4.4 atau *kitkat*. Untuk lebih memperluas jangkauan pengguna, kedepannya perlu dibuat aplikasi dengan sistem berbasis iOS (*Apple*) untuk dapat digunakan pada pengguna yang memiliki *smartphone* berbasis iOS tersebut.

DAFTAR PUSTAKA

- Abduh, A. A. (2018). *Berubah dari Dalam*. Jakarta: PT Serambi Semesta Distribus.
- Abdullah, D., Rahim, R., Siahaan, A. P. U., Ulva, A. F., Fitri, Z., Malahayati, M., & Harun, H. (2018, June). Super-encryption cryptography with IDEA and WAKE algorithm. In *Journal of Physics: Conference Series* (Vol. 1019, No. 1, p. 012039). IOP Publishing.
- Clear, J. (2018). *Atomic Habits*. Jakarta: PT Gramedia Pustaka Utama.
- Darmawan, E. (2015). *Pemrograman Dasar C – JAVA – C# Yang Susah Jadi Mudah!!* Bandung: Informatika.
- Fachri, B., & Surbakti, R. W. (2021). Perancangan Sistem Dan Desain Undangan Digital Menggunakan Metode Waterfall Berbasis Website (Studi Kasus: Asco Jaya). *JOURNAL OF SCIENCE AND SOCIAL RESEARCH*, 4(3), 263-267.
- Faris, Mohammed. (2017). *Muslim Produktif*. Jakarta: PT Elex Media Komputindo.
- Gunawan, G., Nuriyanto, H., Sriadhi, S., Fauzi, A., Usman, A., Fadlina, F., ... & Rahim, R. (2018, June). Mobile application detection of road damage using canny algorithm. In *Journal of Physics: Conference Series* (Vol. 1019, No. 1, p. 012035). IOP Publishing.
- Hakim, L. (2018). *Bahasa Pemrograman*. Yogyakarta: Deepublish.
- Herlina, S. M., & Musliadi KH, S. (2019). *Pemrograman Aplikasi Android dengan Android Studio, Photoshop, dan Audition*. Jakarta: PT Elex Media Komputindo.
- Ismayani, A. (2018). *Cara Mudah Membuat Aplikasi Pembelajaran Berbasis Android dengan Thinkable*. Jakarta: PT Elex Media Komputindo.
- Indrawan, M. I., Alamsyah, B., Fatmawati, I., Indira, S. S., Nita, S., Siregar, M., ... & Tarigan, A. S. P. (2019, March). UNPAB Lecturer Assessment and Performance Model based on Indonesia Science and Technology Index. In *Journal of Physics: Conference Series* (Vol. 1175, No. 1, p. 012268). IOP Publishing.
- Mulyani, S. (2016). *Analisa Dan Perancangan Sistem Informasi Manajemen Keuangan Daerah*. Bandung: Abdi Sistematika.
- Muslihudin, M., & Oktafianto. (2016). *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta: CV Andi Offset.
- Rosa, A., & Shalahuddin, M. (2015). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Strauss, D. (2020). *Getting Started with Visual Studio 2019: Learning and Implementing New Features*. New York: Apress Media.

Suardhika, G. (2015, September 1). Apa itu Produktivitas Diri? *Apa itu Produktivitas Diri?*