

BAB 3

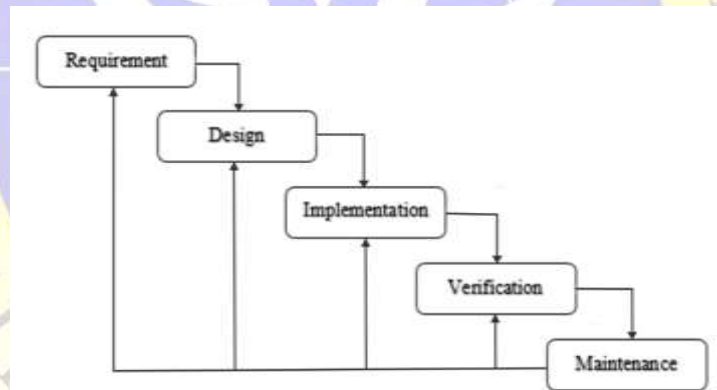
METODE PENELITIAN

3.1 Model Penelitian Dan Pengembangan

Model perbaikan yang digunakan pada tahap pengembangan ini adalah Model *Waterfall*. Untuk alasan apa model *cascade* digunakan pada tahap perbaikan ini, karena model *cascade advance* dianggap lebih mumpuni sehingga dapat mencapai tujuan normal dengan mudahnya setiap tahap, dimulai dengan penentuan *klien* dan kemudian dilanjutkan ke tahap berikut ini:

- a. Perencanaan (*plaining*)
- b. Permodelan (*modeling*), konstruksi (*construction*), serta
- c. Penyerahan sistem ke para pelanggan/pengguna (*deployment*).

Kemudian diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (Pressman,2012) Tahapan metode *waterfall* dapat dilihat pada gambar 3.1 dibawah ini :



Gambar 3.1

Tahapan Metode *Waterfall*

3.2 Prosedur Penelitian dan Pengembangan

Dalam pengembangan metode *waterfall* ini memiliki beberapa tahapan yang berurut yaitu, *Requirement* (analisi kebutuhan), *Design system* (desain sistem), *Coding* (pengkodean), dan *Testing* (pengujian), Penerapan program, pemeliharaan. Tahapan-tahapan dari metode *waterfall* ini adalah sebagai berikut:

3.2.1 Requirement Analisis

Pada tahap ini, pengembangan sistem membutuhkan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh *klien* dan batasan produk. Data ini sebagian besar dapat diperoleh melalui rapat, percakapan, atau tinjauan observasi langsung. Data diselidiki untuk mendapatkan informasi yang dibutuhkan oleh *klien*.

3.2.2 System Design

Rincian prasyarat dari tahap sebelumnya akan dikonsentrasikan pada tahap ini, dan konfigurasi kerangka kerja sudah siap. Kerangka *desain* membantu dalam memutuskan peralatan dan prasyarat kerangka kerja. Dan selanjutnya membantu dalam mendefinisikan rekayasa kerangka umum.

3.2.3 Implementation

Pada tahap ini, kerangka pertama dibentuk menjadi *unit-unit* kecil yang disesuaikan, yang digabungkan pada tahap berikutnya. Setiap *unit* dibuat dan dicoba untuk kegunaan yang dikenal sebagai pengujian *unit*.

3.2.4 Integration And Testing

Semua *unit* yang dibuat pada tahap eksekusi dimasukkan ke dalam kerangka kerja setelah pengujian dilakukan oleh setiap *unit*. Setelah mencampur seluruh kerangka dicoba untuk memeriksa kegagalan atau kesalahan.

3.2.5 Operation And Maintenance

Tahap terakhir dalam model *waterfall*. Pemrograman yang telah dilakukan, dijalankan dan diselesaikan mendukung. Pemeliharaan mencakup memperbaiki kesalahan yang tidak ditemukan pada Langkah sebelumnya. Peningkatan pelaksanaan *unit* kerangka kerja dan peningkatan administrasi kerangka kerja sebagai kebutuhan baru.

3.3 Rekayasa Kebutuhan (*Requirement*).

3.3.1 Kebutuhan Fungsional

Kebutuhan fungsional pada aplikasi ini adalah untuk mempermudah wali santri dalam mendapatkan informasi tentang putra putrinya yang ada di pesantren khususnya pada bidang ubudiyah dan keamanan serta menambah tingkat efisiensi pada bagian asrama mengenai penataan administrasi di bidangnya.

3.3.2 Kebutuhan Non-Fungsional

Kebutuhan non fungsional menjelaskan kebutuhan-kebutuhan non fungsional dalam sistem informasi meliputi:

1. Kebutuhan Perangkat Keras (*hardware*)

- Pc atau Laptop
- *Processor* minimal Intel (R)core(TM)
- *HardDisk atau SSD* dengan kapasitas minimal 250GB
- *RAM* minimal 4GB
- *Mouse*

2. Kebutuhan Perangkat Lunak (*Software*)

Penelitian ini menggunakan perangkat lunak berupa :

- Microsoft Windows 10
- Microsoft Office 2016
- Microsoft Visio 2013
- Android studio 6

3.4 Analisis Kebutuhan

Langkah analisis pada tahap pengumpulan data pada aplikasi ini dilakukan secara intens atau secara rinci, dengan demikian kita dapat mengetahui dan mempermudah dalam mengumpulkan data yang kita butuhkan nantinya. Melalui , observasi, wawancara dan studi pustaka.

3.4.1 Observasi

Adapun bahan observasi dalam teknik pengumpulan data ini adalah pengolahan data buku harian santri yang berada di Pondok Pesantren Salafiyah Syafi'iyah Sukorejo. Metode pencatatan dan pelaporan yang telah berjalan saat ini dengan melalui metode ketik manual dan laporan bulanan yang dirapatkan dengan pengasuh dan para umamak pesantren.

Tabel 3.1 Kegiatan Observasi

waktu	Bagian	Uraian Kegiatan	Target Pencapaian
08 September 2021.	Staff Asrama	1. Melihat isi buku harian santri 2. Menanyakan proses pengisian buku	1. Untuk mengetahui laporan di buku 2. Untuk mengetahui apa saja yang dicatat pada buku harian santri
09 September 2021	Staff bagian ubudiyah pesantren	1. Mengamati kegiatan apa saja yang dipantau oleh pihak ubudiyah	1. Untuk mengetahui kendala apa saja yang seharusnya dilaporkan
09 September 2021	Santri yang dipanggil pihak keamanan	1. Mengamati pelanggaran yang sering terjadi	1. Untuk mengetahui pelanggaran yang akan dilaporkan

3.4.2 Wawancara

Wawancara ini dilakukan terhadap ketua kamar dan staff kepesantrenan bagian asrama. Adapun sesi wawancara yg didapat sebagai berikut:

Tabel 3.2 Kegiatan Wawancara

No	Tanggal	Bagian	Pertanyaan	Target Pencapaian
1	8 September 2021	1. Staff kepesantrenan bagian asrama	1. Apa saja kegiatan para santri yang dilaporkan ketika rapat bulanan dengan pengasuh	1. Dapat mengetahui output dari hasil pencatatan bagian asrama

Tabel 3.3 Lanjutan

No	Tanggal	Bagian	Pertanyaan	Target Pencapaian
		2. Ketua kamar	2. Bagaimana proses pengisian buku harian santri	2. Mengetahui prosedur pengisian buku harian santri

3.5 Studi Pustaka

Untuk menunjang penelitian ini agar lebih luas dalam pengkajiannya yaitu dengan mempelajari berbagai sumber yang sudah tersedia, baik itu dari internet, buku journal, koran, artikel dan dari beberapa sumber-sumber yang lainnya, sehingga dapat mencapai hasil yang lebih efisien.

3.6 Desain

Setelah menganalisa permasalahan di Pondok Pesantren Salafiyah Syafi'iyah Sukorejo husunya bagian Asrama tentang buku harian santri, pada langkah ini adalah mendesain aplikasi serta menjelaskan tentang desain pada aplikasi Buku Harian Santri dengan menggunakan Android, yaitu proses multi langkah yang fokus pada desain pembuatan aplikasi Buku Harian Santri sekiranya memungkinkan wali santri untuk mendapatkan akses informasi terhadap putra putrinya yang berada di pesantren. Pada penelitian ini, desain aplikasi dilakukan dengan merancang aplikasi menggunakan metode *flowchart* serta desain aplikasinya.

3.7 Penulisan Program

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, dalam pembuatan Aplikasi Buku Harian Santri, aplikasi ini akan dituangkan kedalam bahasa pemrograman android melalui proses coding. Tahap ini merupakan implementasi dari tahap *design* yang secara teknis nantinya dikerjakan oleh programmer.

3.8 Pengujian

Proses uji coba keberhasilan perancangan dan implementasi yang menggunakan *Black Box testing*. Tujuannya untuk menganalisis aplikasi jika terdapat kekurangan sehingga aplikasi ini perlu adanya perbaikan. *Black box testing* adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak agar software bebas dari error, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

Adapun fungsi yang akan diuji sebagai berikut :

1. Halaman awal
2. Menu utama
3. Menampilkan data santri
4. Menampilkan laporan kegiatan santri

a. Pengujian Internal (*Black Box*)

Pengujian perangkat lunak mengenai aplikasi ini menggunakan data uji berupa *elemen buton* dari aplikasi perangkat lunak yang telah dibuat dengan *Blackbox*. *Black box* merupakan pengujian yang dilakukan hanya mengamati hasil eksekusi data uji dan memeriksa fungsional dari perangkat. *Black box testing* berfokus pada persyaratan fungsional perangkat lunak. Metode ini memungkinkan *software developer* untuk mendapatkan serangkaian kondisi *input* yang mempergunakan semua persyaratan fungsional program. Fungsional dari *black box* adalah melakukan pengujian berdasarkan apa yang dilihat, hanya fokus terhadap fungsionalitas dan *output*. Pengujian lebih ditunjukkan pada desain *software* sesuai standar. Berikut rencana pengujian pada aplikasi yang akan dilakukan.

Tabel 3.4 Pengujian Internal (*Black Box Testing*).

Objek		Event		Keterangan
-------	--	-------	--	------------

	Fungsi		Hasil yang diharapkan	Sesuai	Belum Sesuai
Login	Untuk diklik	Untuk Masuk ke menu	Menampilkan menu utama		
Tombol identitas santri	Untuk diklik	Untuk identitas santri	Menampilkan identitas lengkap santri		
Tombol keamanan	Untuk diklik	Untuk data keamanan	Menampilkan pelanggaran yang dilakukan santri		
Tombol ubudiyah	Untuk diklik	Untuk data Ubudiyah	Menampilkan pelanggaran santri dibidang ubudiyah		

Hasil Pengujian Langsung (pengujian Langsung Kepada *User*) *unit* testing dalam pengujian pada program tersebut lebih menekankan pada pengujian secara *BlackBox*, sistem testing wawancara yang dilakukan setelah pengujian program terhadap *user* adalah sebagai berikut observasi dan wawancara pengujian eksternal.

3.9 Pengujian Eksternal

Dalam mengamati fungsional dan keluaran (*output*) pengujian sistem. Pengujian pada tahap eksternal dilakukan dengan menggunakan angket. Angket berisi pertanyaan-pertanyaan seputar sistem yang sudah dibangun, apakah hasilnya sudah sesuai dengan kebutuhan yang diharapkan. Adapun pengujian eksternal terdapat pada keterangan tabel berikut:

Tabel 3.5 Pengujian Eksternal

NO.	Subyek	Pertanyaan	Hasil				
			SB	B	C	K	SK
1	Staff Kepesantrenan Bagian Asrama	apakah sistem saat ini berjalan sesuai harapan?					
2	Staff Kepesantrenan Bagian Asrama	Apakah sistem saat ini mudah untuk dioperasikan?					
3	Staff Kepesantrenan Bagian Asrama	Apakah sistem saat ini layak untuk digunakan?					
4	Staff Kepesantrenan Bagian Asrama	Apakah fungsi dalam sistem saat ini berjalan sesuai yang diharapkan?					
5	Staff Kepesantrenan Bagian Asrama	Apakah desain dari sistem ini sudah menarik?					

Ditahap ini responden dapat menjawab pertanyaan sesuai dengan jawaban yang telah disediakan. Adapun table keterangan serta bobot dari setiap pertanyaan dalam angket ialah berikut ini:

Tabel 3.6 Keterangan dan Bobot Nilai

No	Keterangan	Bobot Nilai
1	SB = Sangat Baik	50
2	B = Baik	40

Tabel 3.7 Lanjutan

No	Keterangan	Bobot Nilai
3	C = Cukup	30
4	K = Kurang	20
5	SK = Sangat Kurang	10

Untuk mengetahui hasil dari angket diatas, terlebih dahulu harus mengetahui nilai persentase dari setiap jawaban. Untuk mengetahui interval hal yang harus dilakukan terlebih dahulu ialah mencari interval skor persen (i) menggunakan rumus sebagai berikut:

$$I = 100 / \text{jumlah skor (likert)}$$

Jika jumlah skor 20, maka $100/20 = 50$

Hasil (I) = 50

Jadi jarak intervalnya adalah 50%

Adapun table interpretasi skor persen berdasarkan perhitungan interval ialah sebagai berikut:

Tabel 3.8 Keterangan dan Nilai Persentase Skor Interval

No	Keterangan	Nilai Persentase
1	SK = Sangat Kurang	0% - 19,99%
2	K = Kurang	20% - 39,99%
3	C = Cukup	40% - 59,99%
4	B = Baik	60% - 79,99%
5	SB = Sangat Baik	80% - 100%

Untuk mengetahui jumlah jawaban yang diperoleh dari tahapan pengujian eksternal diatas dapat diketahui melalui rumus sebagai berikut:

Rumus: $T \times P_n$

Keterangan **T**: total jumlah responden yang memilih

P_n: Pilihan angka skor likert

Selanjutnya adalah mencari interpretasi perhitungan skor dengan cara mencari nilai terkecil (Y) dan tertinggi (X) skor likert.

Y = skor tertinggi likert x jumlah responden

X = skor terendah likert x jumlah responden

Setelah tahapan diatas selesai maka tahap terakhir dalam mengetahui hasil interpretasi responden terhadap layak dan tidaknya aplikasi tersebut adalah melihat hasil nilai yang dihasilkan dengan menggunakan rumus $\text{Index \%} = \text{Total Skor} / Y \times 100$.

3.10 Operation dan Maintenance

Tahap akhir dalam model *waterfall*. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

Pemeliharaan suatu *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada error kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengemangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergatian sistem operasi, atau perangkat lainnya.

Oleh karena itu aplikasi ini perlu adanya update fitur baru seiring perkembangannya zaman dan pemeliharaan (perawatan) sehingga dapat mengantisipasi terjadinya eror ketika aplikasi digunakan.