

**ANALISIS DAN PERANCANGAN APLIKASI *POINT OF SALE* BERBASIS
MOBILE MENGGUNAKAN *FRAMEWORK FLUTTER* DENGAN METODE *RAPID
APPLICATION DEVELOPMENT (RAD)***

***ANALYSIS AND DESIGN OF MOBILE-BASED POINT OF SALE APPLICATIONS
(POS) USING THE FLUTTER FRAMEWORK WITH THE RAPID APPLICATION
DEVELOPMENT (RAD) METHODOLOGY***

Marco Alfian Sumarto

Balai Pengembangan Sumber Daya Manusia dan Penelitian Komunikasi dan Informatika Jakarta
Jalan Pegangsaan Timur No.19B, Cikini, Menteng, RT.1/RW.1, Pegangsaan, Kec. Menteng, Kota Jakarta
Pusat, Daerah Khusus Ibukota Jakarta, Indonesia
marc004@kominfo.go.id

Diterima tgl. 07/04/2023; Direvisi tgl. 28/06/2023; Disetujui tgl. 30/06/2023

ABSTRACT

Micro, Small, and Medium Enterprises (MSMEs), also known as UMKM in Indonesia, have a dominant number in the country. One of the challenges faced by MSMEs today is difficulty in managing their sales transactions effectively. There are still MSMEs that have trouble calculating transactions quickly and accurately, have poor inventory management, and require a lot of time to produce transaction reports. Tofan is one of the MSMEs that facing these problems. One solution to this problem is to utilize a point of sales (POS) application, but unfortunately Tofan Computer does not have sufficient resources to develop the application, especially in identifying needs and determining the application design. This study analyzes and designs POS application systems to solve problems in these SMEs with Tofan Computer as a case study. The method used follows the Rapid Application Development (RAD) approach. From this research, 22 system requirements were generated which were divided into functional and non-functional requirements. Based on the requirements of the system, architecture, interface, and database designs were also made. A POS application prototype was then developed based on this design using the Dart programming language with the Flutter framework. This POS application has also been tested using black box testing techniques and the results show that the application can run well and is responsive. The RAD method can speed up the POS application development process and increase flexibility in accommodating the evolving needs of users.

Keywords: *Rapid Application Development (RAD), micro, small, and medium enterprises (MSMEs), Point of Sale (POS) application, user requirements analysis*

ABSTRAK

Usaha Mikro, Kecil, dan Menengah (UMKM) memiliki jumlah yang dominan di Indonesia. Salah satu permasalahan yang dihadapi oleh UMKM saat ini adalah kesulitan dalam mengelola transaksi penjualannya dengan baik. Masih ditemukan UMKM yang kesulitan menghitung transaksi dengan cepat dan akurat, memiliki manajemen stok yang kurang baik, dan membutuhkan banyak waktu untuk menghasilkan laporan transaksi. Tofan *Computer* adalah salah satu UMKM yang menghadapi permasalahan-permasalahan tersebut. Solusi untuk permasalahan tersebut salah satunya adalah dengan memanfaatkan aplikasi *point of sales* (POS), namun sayangnya Tofan *Computer* tidak memiliki sumber daya yang cukup untuk mengembangkan aplikasi tersebut terutama dalam mengidentifikasi kebutuhan dan menentukan rancangan aplikasinya. Penelitian ini melakukan analisis dan perancangan sistem aplikasi POS untuk menyelesaikan permasalahan pada UMKM tersebut dengan Tofan *Computer* sebagai studi kasusnya. Metode yang digunakan mengikuti pendekatan *Rapid Application Development (RAD)*. Dari penelitian ini dihasilkan sebanyak 22 kebutuhan sistem yang terbagi menjadi kebutuhan fungsional dan nonfungsional. Dari kebutuhan sistem tersebut juga berhasil dibuat rancangan arsitektur, antarmuka, serta basis data. Sebuah purwarupa aplikasi POS kemudian dikembangkan berdasarkan rancangan tersebut menggunakan bahasa pemrograman Dart dengan Flutter *framework*. Aplikasi POS ini juga telah diuji coba menggunakan teknik *black box testing* dan hasilnya menunjukkan bahwa aplikasi dapat berjalan dengan baik dan responsif. Metode RAD dapat mempercepat proses pengembangan aplikasi POS dan meningkatkan fleksibilitas dalam mengakomodasi kebutuhan pengguna yang terus berkembang.

Kata Kunci: *Rapid Application Development (RAD), usaha mikro, kecil, dan menengah (UMKM), aplikasi Point of Sale (POS), analisis kebutuhan pengguna*

1. PENDAHULUAN

Usaha mikro, kecil, dan menengah atau yang lebih dikenal dengan UMKM merupakan entitas usaha yang paling banyak dibanding unit usaha lain di Indonesia. Berdasarkan data dari Kementerian Koperasi dan Usaha Kecil dan Menengah, jumlah UMKM nasional yang tercatat sudah mencapai 64,2 juta unit usaha (Kementerian Koperasi dan UKM, 2018). Namun, masih banyak pelaku UMKM yang memiliki keterbatasan pada sumber daya, proses dan prosedur yang belum ada. Permasalahan UMKM yang sering dihadapi saat ini salah satunya adalah pengelolaan transaksi penjualan. Tidak sedikit pelaku UMKM di Indonesia yang tidak memiliki prosedur yang baku pada transaksi, pembukuan yang belum jelas, dan proses yang masih manual (Syafaruddin, S., & Hamidah, N., 2020).

Pengelolaan transaksi penjualan dengan baik menjadi hal penting agar dapat mengetahui persediaan barang secara akurat, melihat laporan secara tepat waktu dan juga sangat memengaruhi perkembangan usaha. Saat ini banyak UMKM yang kesulitan mengetahui jumlah aset yang dimiliki, karena tidak mengelola transaksi dengan benar sehingga menjadi sulit berkembang (Badan Pusat Statistik, 2020). Seiring berkembangnya zaman yang semakin modern, permasalahan seperti pengelolaan transaksi usaha dapat dibantu dengan memanfaatkan kemajuan teknologi informasi. Aplikasi POS (*Point of Sale*) adalah aplikasi yang dapat membantu bagian transaksi penjualan khususnya pada perusahaan retail (Putra, B. A., 2018). Karena adanya kebutuhan untuk membuat proses penjualan menjadi praktis dan cepat sehingga menghindari pelanggan untuk antri dalam waktu yang lama. Mokapos, Pawoon, dan Olsera adalah aplikasi POS yang biasa digunakan. Pelaku usaha dapat menggunakan sistem ini untuk memudahkan kasir dalam mengelola proses transaksi penjualan.

Usaha mikro memiliki kesulitan dalam mengadopsi aplikasi POS yang sudah ada. Salah satu kesulitannya yaitu untuk membayar biaya langganan pada aplikasi tersebut. Karena usaha mikro memiliki keterbatasan seperti modal usaha, sumber daya manusia yang sedikit dan volume transaksi yang masih kecil. Sedangkan, untuk membuat sistem pengelolaan transaksi penjualan UMKM dengan mengembangkan aplikasi POS bukan hal mudah (Nuryantono, N., & Kristiani, E., 2019). Proyek pengembangan sistem informasi perlu melewati tahapan yang rumit, mulai dari *planning, analysis, design* dan *implementation*. Semua proyek membutuhkan analisis untuk mengumpulkan persyaratan, memodelkan kebutuhan bisnis, dan membuat cetak biru tentang bagaimana sistem harus dibangun (A Dennis, 2012).

Penggunaan perangkat *mobile* memiliki catatan penggunaan masyarakat yang paling luas di Indonesia. Mengutip data We Are Social dan Hootsuite (2020), pada awal tahun 2020 populasi pengguna perangkat *mobile* di dunia sangat tinggi hingga mencapai 5,19 miliar pengguna, dengan tingkat penetrasi mencapai 67% populasi dan penambahan 124 juta pengguna dibandingkan tahun sebelumnya. Tingkat penggunaan perangkat *mobile* di Indonesia sebesar 338,2 juta lebih banyak 124% dibandingkan jumlah populasi di Indonesia (We Are Social & Hootsuite, 2020). Data tersebut menjelaskan setiap individu di Indonesia memiliki lebih dari satu perangkat *mobile*, bahkan penggunaan perangkat *mobile* tidak hanya digunakan secara luas oleh masyarakat, tetapi juga digunakan oleh pelaku usaha (We Are Social & Hootsuite, 2020). *Smartphone* adalah perangkat *mobile* yang sudah terjangkau bahkan bagi usaha mikro. Menggunakan *smartphone* sebagai pengganti mesin kasir akan sangat membantu mengelola transaksi penjualan.

Smartphone saat ini memiliki banyak sistem operasi, tetapi dua sistem operasi yang menguasai pasaran dunia *smartphone* diantaranya yaitu Android dan iOS. Untuk membuat aplikasi POS yang dapat bekerja pada Android dan iOS akan banyak memakan waktu jika membuat dua aplikasi *mobile* secara terpisah. Kekhawatiran terbesar mengembangkan aplikasi *mobile* adalah

performance, flutter mengklaim sudah memiliki *full native performance* yang menyesuaikan dengan IOS dan Android hanya dengan satu *codebase*. Dikutip dari situs resminya *flutter* merupakan *framework* pengembangan aplikasi *mobile*, *web* dan *desktop* yang dikembangkan oleh Google (*Flutter.dev*, 2020).

Menurut Nuryantono (2019), aplikasi POS yang sudah ada memiliki fitur-fitur yang tidak sesuai dengan kebutuhan dan skala usaha mikro, serta kemampuan untuk mengelola inventaris atau pelaporan yang terlalu kompleks untuk kebutuhan usaha mikro. Pada usaha mikro perlu penyesuaian fitur aplikasi POS berdasarkan kebutuhan mereka dan hal ini membutuhkan perencanaan dan perancangan aplikasi POS berdasarkan *user requirement*. Tahap analisis dan desain merupakan tahapan terpenting dan tersulit dari siklus *System Development Life Cycle* (SDLC) dalam pengembangan fitur aplikasi POS. Pada usaha mikro akan sulit melakukan dua hal tersebut yaitu menganalisis kebutuhan dan membuat rancangan aplikasi yang sesuai dengan kondisi UMKM (Nuryantono, 2019). Oleh karena itu, penelitian ini dilakukan untuk mengatasi masalah tersebut dengan mengembangkan aplikasi POS yang efektif dan efisien untuk UMKM dengan metode Rapid Application Development (RAD). Metode RAD akan mempercepat waktu pengembangan aplikasi dan memungkinkan penggunaan aplikasi lebih mudah dan terjangkau bagi UMKM.

Merancang aplikasi *point of sales* yang berjalan di perangkat *mobile* dapat dilakukan dengan mengimplementasikan *Framework Flutter*. Selain itu, aplikasi POS yang tepat dapat membantu meningkatkan kualitas pelayanan dan mengoptimalkan pengelolaan persediaan barang. Hal ini akan berdampak positif pada kemudahan transaksi penjualan pada UMKM di Indonesia, khususnya pada usaha skala mikro.

1.1. Aplikasi *Point of Sale* di Usaha Mikro Kecil Menengah

Point of Sale atau disingkat sebagai POS adalah sebuah sistem aplikasi yang diterapkan pada bisnis pertokoan dalam menangani pengolahan data transaksi seperti data pembelian, penjualan eceran, transaksi hutang, transaksi retur pembelian, dan pelaporan transaksi yang dibutuhkan dalam pengambilan keputusan oleh para pebisnis swalayan, organisasi, atau perusahaan (Sugihartono et al., 2015).

Menurut Sherly (Siregar & Mulyani, 2021), aplikasi POS telah menjadi solusi yang sangat berharga bagi UMKM dalam mengoptimalkan operasional mereka. Dengan menggunakan aplikasi POS, UMKM dapat mengelola proses penjualan dan pengelolaan stok dengan lebih efisien, meningkatkan pengalaman pelanggan, dan menghasilkan data yang berharga untuk pengambilan keputusan bisnis yang lebih baik. Salah satu manfaat utama dari aplikasi POS adalah kemampuannya untuk mencatat transaksi penjualan secara otomatis. Dengan menggunakan perangkat keras yang terhubung dengan aplikasi POS, UMKM dapat dengan cepat memindai produk dan menerima pembayaran dari pelanggan. Data penjualan tersebut langsung tercatat di sistem, menghilangkan kebutuhan untuk mencatat secara manual dan mengurangi risiko kesalahan manusia. Hal ini membantu UMKM menghemat waktu dan tenaga yang dapat dialokasikan untuk aspek bisnis lainnya.

1.2. Rekayasa Perangkat Lunak dengan *Framework Flutter*

Dalam mengembangkan perangkat lunak, perancangan merupakan sesuatu yang penting. Merancang sesuai dengan analisis menjadi titik berat dalam mengembangkan perangkat lunak. Agar menghasilkan perangkat lunak yang sesuai dan berkualitas tinggi sehingga menjadi nilai bisnis bagi penggunanya. Menurut Setiyani, rekayasa perangkat lunak merupakan proses produksi perangkat lunak yang menggunakan pendekatan sistematis, disiplin dan dapat diukur yang pada

akhirnya menghasilkan perangkat lunak yang bekerja secara efisien (Setiyani, 2019). Sedangkan menurut Rosa, rekayasa perangkat lunak merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan untuk menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin (Rosa A. S, 2018).



Sumber: *Software Engineering* (Setiyani, 2019)

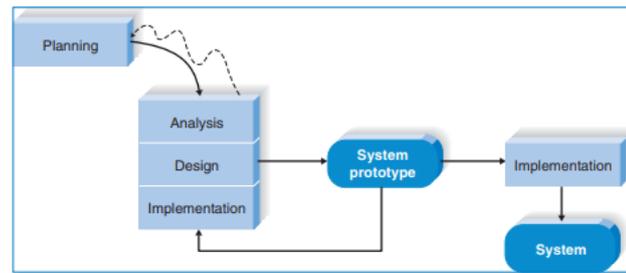
Gambar 1. Lapisan – lapisan rekayasa Perangkat Lunak

Fondasi dari tindakan yang berkaitan dengan rekayasa perangkat lunak adalah lapisan proses. Proses perangkat lunak mendefinisikan kerangka kerja (*framework*) yang harus dibangun, konteks dan pembentukan kendali manajemen proyek perangkat lunak. Lapisan metode berisi tentang prosedur yang mencakup komunikasi, analisis kebutuhan, pemodelan rancangan, pembuatan program, serta pengujian. Lapisan perangkat menyajikan dukungan semi otomatis maupun otomatis bagi proses dan metode-metode (Setiyani, 2019).

Sebagai salah satu contoh *framework* yang dapat digunakan dalam rekayasa perangkat lunak adalah Flutter. Flutter adalah sebuah *framework open-source* yang dikembangkan oleh Google. Flutter memungkinkan pengembangan aplikasi Android dan iOS dengan performa tinggi menggunakan basis kode yang sama (Flutter.dev, 2020). Flutter memiliki beberapa keuntungan dalam membangun aplikasi. Pertama, dengan menggunakan "One Codebase," pengembang dapat menggunakan satu basis kode untuk mengembangkan aplikasi di berbagai platform. Keuntungan kedua adalah "Fast Development" dengan fitur *Hot Reload* yang memungkinkan bereksperimen, membuat UI, menambahkan fitur, dan memperbaiki *bug* dengan cepat. Flutter juga menawarkan antarmuka pengguna yang "Expressive and Flexible" dengan *widget Material Design* dan *Cupertino* yang indah, API gerakan yang kaya, dan pengguliran alami yang mulus. Keuntungan terakhir adalah "Native Performance," di mana *widget* Flutter menggabungkan perbedaan platform penting untuk memberikan kinerja asli di *iOS* dan *Android*. Flutter menggunakan bahasa pemrograman *Dart* yang memungkinkan pembaruan pohon tampilan untuk setiap bingkai baru. Dengan memanfaatkan keuntungan-keuntungan ini, pengembang dapat membangun aplikasi lintas platform dengan antarmuka pengguna yang menarik dan kinerja yang optimal (Flutter.dev, 2020).

1.3. Pengembangan Sistem dengan Metode *Rapid Application Development* (RAD)

Konsep pengembangan sistem dengan metode *Rapid Application Development* (RAD) adalah kumpulan metodologi yang muncul sebagai respon terhadap kelemahan *waterfall development* dan variasinya (A Dennis, 2012a). RAD menggabungkan teknik khusus dan komputer *tools* untuk mempercepat tahap analisis, desain, dan implementasi beberapa bagian dari pengembangan sistem dengan cepat dan sampai ke tangan pengguna untuk evaluasi dan umpan balik.



Sumber: Alan Dennis ,2012

Gambar 2. Prototipe Sistem RAD

Gambar 2 merupakan alur pengembangan sistem dari metode RAD dengan pendekatan prototipe sistem. Penelitian ini menggunakan alur tersebut dalam proses pengembangannya. Mulai dari mengikuti reaksi dan komentar dari pengguna, pengembang menganalisis ulang, mendesain ulang, dan memperbaiki kekurangan dan menambahkan lebih banyak fitur dengan prototipe kedua. Tahap analisis ini berlanjut sampai dengan pengguna setuju bahwa prototipe menyediakan fungsionalitas yang cukup untuk dipasang dan digunakan di organisasi. Pendekatan ini sangat berguna ketika pengguna mengalami kesulitan dalam mengungkapkan persyaratan untuk sistem.

Secara umum, kerangka penelitian ini melalui tahap analisis dan desain secara bersamaan untuk pengembangan versi sederhana dari sistem yang diusulkan dengan bentuk prototipe sistem. Kemudian, prototipe yang telah dibuat tersebut diberikan kepada pengguna untuk evaluasi dan umpan balik. Sampai pengguna setuju dengan sistem yang diusulkan, maka dapat berlanjut ke tahap berikutnya.

2. METODE PENELITIAN

Jenis penelitian yang digunakan dalam penelitian ini adalah studi kasus. Studi kasus merupakan metode menghimpun dan menganalisis data berkenaan dengan adanya masalah, kesulitan, hambatan, dan penyimpangan (Ahyar et al., 2020). Tetapi dapat juga dijadikan kasus karena keunggulan dan keberhasilannya. Studi kasus menghasilkan data untuk selanjutnya dianalisis untuk menghasilkan teori. Seperti lazimnya perolehan data dalam penelitian kualitatif, data studi kasus diperoleh dari beberapa teknik, seperti wawancara, observasi, dan dokumentasi.

Fenomena yang dijadikan kasus dalam penelitian ini adalah pengelolaan transaksi penjualan yang kurang baik sehingga menghambat perkembangan bisnis pada salah satu UMKM di kota Cikarang yaitu UMKM *Tofan Computer*. *Tofan Computer* merupakan salah satu UMKM yang bergerak dibidang pengadaan komputer, perangkat keras, perangkat lunak dan jasa *Information Technology Solution* lainnya. Dalam peningkatan pelayanan dan daya saing, *Tofan Computer* membutuhkan aplikasi untuk membantu mengelola transaksi penjualannya. UMKM ini memiliki jumlah 4 pegawai yang terdiri dari teknisi dan penjaga toko atau penjual. Penelitian dilakukan dengan pendekatan analisis sistem transaksi penjualan yang ada, agar dapat merancang perangkat lunak yang dapat membantu kebutuhan bisnis pada UMKM *Tofan Computer*.

Alur pengembangan sistem diuraikan menjadi beberapa langkah yang sistematis untuk memperoleh jawaban dan mencapai tujuan penelitian yang telah ditetapkan. Langkah-langkah tersebut menjelaskan tentang metode atau proses apa saja yang dilakukan, *input* yang dibutuhkan serta *output* yang dihasilkan (Tabel 1).

Tabel 1. Langkah Penelitian

No.	Langkah penelitian	Metode / Proses	Input	Output
1	Merumuskan Masalah	Wawancara dan Observasi	Kondisi saat ini dari tempat studi kasus	Pertanyaan penelitian
2	Mengumpulkan <i>Requirements</i>	Wawancara, Observasi, Studi Literatur dan <i>Best Practice</i>	Pertanyaan penelitian	Daftar <i>requirements: functional & non-functional req</i>
3	Memodelkan <i>Requirements</i>	<i>Functional Modeling, Process Modeling, Data Modeling</i>	Daftar <i>requirements</i>	<i>Requirements model: UML, DFD</i>
4	Membuat Rancangan Perangkat Lunak	<i>Designing Information Architecture, Designing Interface, Designing Databases</i>	<i>Requirements model</i>	<i>Design model: Sitemap, Mockup, Prototipe</i>
5	Mengembangkan Purwarupa	<i>Coding</i>	<i>Design model</i>	Purwarupa aplikasi
6	Menguji Purwarupa	<i>Unit Testing, Usability Testing</i>	Purwarupa aplikasi	Hasil pengujian
7	Menyusun Kesimpulan dan Rekomendasi	Analisis Hasil Pengujian	Hasil pengujian	Kesimpulan dan rekomendasi

Sumber: Hasil Olah Data Penulis, 2023

Tabel 1 merupakan langkah penelitian yang digunakan dalam penelitian ini untuk mencapai tujuan penelitian. *Input* (masukan) dan *Output* (hasil) dari langkah-langkah tersebut akan dijelaskan secara eksplisit, sehingga maksudnya dapat dipahami dengan mudah.

Sebelum melakukan pengembangan sistem, perlu dilakukan identifikasi proses dan permasalahan di tahapan POS pada UMKM tersebut. Langkah ini sangat penting untuk menentukan langkah-langkah penelitian atau pengembangan selanjutnya. Dalam konteks ini, identifikasi proses bisnis UMKM akan membantu dalam memahami bagaimana UMKM tersebut menjalankan operasionalnya, termasuk tahapan POS. Kemudian melakukan observasi terhadap pelaksanaan transaksi penjualan yang ada. Selanjutnya melakukan pengumpulan *requirements* melalui beberapa metode pengumpulan data. Metode pengumpulan data yang digunakan dalam penelitian ini, antara lain:

a) Metode Wawancara

Metode ini dilakukan melalui sesi tanya jawab. Untuk mendapatkan segala informasi yang dibutuhkan dengan memberikan beberapa pertanyaan secara langsung kepada salah satu pemilik UMKM. Hasil dari metode ini adalah daftar kebutuhan atau persyaratan pengguna.

b) Metode Observasi

Metode ini dilakukan melalui pengamatan secara langsung terhadap bagaimana proses pelayanan, melakukan transaksi penjualan hingga laporan penjualan yang digunakan agar menjadi masukan untuk kepada penulis. Hasil yang di dapat dari metode ini adalah daftar kebutuhan atau persyaratan bisnis.

c) Metode Studi Pustaka dan *Best Practice*

Metode ini dilakukan melalui riset studi pustaka dengan cara menelaah buku-buku, literatur-literatur dan beberapa sumber lainnya berupa data yang berhubungan dengan penelitian serupa. Metode ini dilakukan untuk mendapatkan kelengkapan informasi yang dapat mendukung hasil laporan penelitian sesuai permasalahan dan topik yang dibahas.

Data yang telah dikumpulkan dari beberapa metode tersebut berguna untuk menentukan perancangan sistem atau perangkat lunak yang ideal. Kebutuhan atau persyaratan bisnis (*business requirements*) dan kebutuhan atau persyaratan pengguna (*user requirements*) dijadikan acuan dalam membuat kebutuhan perangkat lunak (*software requirements*). Kebutuhan perangkat lunak dalam penelitian ini dibagi menjadi kebutuhan fungsional (*functional requirements*) dan kebutuhan non-fungsional (*non-functional requirements*). Dalam penelitian ini, langkah mengumpulkan *requirements* dan beberapa langkah selanjutnya dilakukan secara berulang. Sampai pengembang dan pengguna setuju bahwa prototipe yang dirancang sudah menyediakan fungsionalitas yang cukup untuk dipasang dan digunakan.

Memodelkan *requirements* bertujuan membantu pengguna dan penulis untuk memahami rancangan sebuah sistem. Dalam penelitian ini menggunakan beberapa metode memodelkan *requirements* antara lain *functional modeling*, *process modeling* dan *data modeling*. Selanjutnya yaitu membuat rancangan perangkat lunak, langkah ini menentukan bagaimana sistem akan beroperasi dalam perangkat keras. Serta bagaimana antarmuka pengguna, formulir, dan laporan yang akan digunakan. Semua bagian tersebut dirancang berdasarkan *requirements* model yang telah dibuat. Selanjutnya dilakukan pengembangan purwarupa di mana rancangan yang sudah dibuat dikembangkan menjadi purwarupa aplikasi dengan melakukan *coding* sesuai dengan prototipe dari desain yang sudah dibuat.

Menguji purwarupa merupakan bagian dari tahap implementasi. Langkah ini bertujuan menguji aplikasi yang sudah dikembangkan, apakah sudah memenuhi persyaratan yang telah ditetapkan dan juga menentukan perbedaan antara hasil yang diharapkan dengan hasil sebenarnya. Pada penelitian ini dilakukan dua jenis pengujian yaitu *usability testing* dan *unit testing*. Untuk dapat melakukan pengujian dengan benar dan dapat dipertanggungjawabkan, maka harus disusun skenario untuk pengujian terlebih dahulu. Skenario pengujian adalah dokumen yang berisi persiapan dan langkah-langkah yang harus dilakukan untuk menguji aplikasi secara terinci. Menyusun kesimpulan dan rekomendasi merupakan langkah terakhir dalam penelitian ini. Pada langkah ini, hasil dari pengujian yang telah dilakukan di analisis kembali. Analisis tersebut bertujuan untuk mengambil kesimpulan dari apa yang telah diujikan. Semua kekurangan maupun masukan yang dapat meningkatkan kualitas perancangan dicatat dan disusun sehingga dapat diperoleh kesimpulan dan rekomendasi yang baik bagi dalam pengembangan aplikasi POS ini.

Alat yang digunakan dalam pengembangan aplikasi meliputi perangkat keras dan perangkat lunak. Perangkat keras yang digunakan dalam penelitian ini adalah laptop HP Probook 450 G5 dan *smartphone* Advan G5 dengan spesifikasi tertentu. Sedangkan perangkat lunak yang digunakan adalah Windows 10, Visual Studio Code versi 1.47.0, Flutter SDK versi 1.12.13+hotfix.8, Java SDK version 8 *Update 221*, dan Figma.

Prosedur pengembangan aplikasi dalam penelitian ini meliputi urutan proses penjualan yang dilakukan oleh UMKM, identifikasi data produk yang dijual, identifikasi hambatan dalam tiap langkah prosedur penjualan, identifikasi tipe laporan yang diperlukan terkait sistem penjualan tunai, identifikasi masalah dalam sistem penjualan tunai saat ini, pengumpulan daftar pegawai dan outlet, pengumpulan kemungkinan pemecahan masalah dari para pelaksana, dan penetapan jadwal waktu analisis, perancangan, dan implementasi sistem. Fokus pengembangan yang dilakukan menitik beratkan pada proses bisnis transaksi penjualan pada usaha mikro.

3. HASIL DAN PEMBAHASAN

Analisis dan perancangan pengembangan sistem POS meliputi tahapan identifikasi *requirement*, pemodelan *requirement*, membuat rancangan, mengembangkan purwarupa, dan

pengujian purwarupa. Pengembangan dengan RAD yang dilakukan memberikan efisiensi waktu pada proses pengembangan sistem POS. Tahap identifikasi *requirement* dilakukan dengan melakukan pengumpulan informasi dari berbagai pihak terkait, seperti pengguna akhir dan pemilik bisnis. Setelah itu, tahap pemodelan *requirement* dilakukan untuk menggambarkan secara rinci kebutuhan sistem POS yang akan dikembangkan.

Pada pengembangan aplikasi POS menggunakan *Framework Flutter* dengan metode RAD sehingga waktu yang dibutuhkan dalam proses pengembangan dapat dihemat. Tim pengembang dapat secara cepat merespon perubahan dan umpan balik dari pengguna akhir, sehingga sistem dapat segera diperbaiki atau disesuaikan dengan kebutuhan yang sebenarnya. Hal ini berkontribusi pada efisiensi waktu dan pengembangan yang lebih optimal.

3.1 Mengumpulkan *Requirements*

Business Requirements atau persyaratan bisnis pada penelitian ini diperoleh dari metode observasi yang telah dilakukan dan divalidasi dengan melakukan sesi wawancara dengan salah satu pemilik UMKM Tofan *Computer*. Selain mengelola dan mendokumentasikan transaksi penjualan, perusahaan juga perlu mengelola dan mengetahui biaya operasional; perlu mengetahui efisiensi kinerja pegawai; perlu mengetahui laporan transaksi penjualan; perlu mendokumentasikan produk yang berupa dari merek produk, detail produk, harga produk, stok produk. Berdasarkan data tersebut, penelitian ini mengikhtisarkan menjadi sebuah daftar kebutuhan bisnis. Daftar tersebut berisikan apa saja hal yang dibutuhkan oleh UMKM Tofan *Computer* agar proses bisnis dapat berjalan dengan baik (Tabel 2).

Tabel 2. Daftar Kebutuhan Bisnis

Daftar Kebutuhan Bisnis (<i>Business Requirements</i>)		
No.	Uraian	Kode
1.	Perusahaan perlu mengelola dan mendokumentasikan transaksi penjualan	BR01
2.	Perusahaan perlu mengetahui laporan transaksi penjualan	BR02
3.	Perusahaan perlu mendokumentasikan produk yang berupa (merk produk, detail produk, harga produk, stok produk)	BR03

Sumber: Hasil Olah Data Penulis, 2023

User requirements atau persyaratan pengguna diperoleh dari hasil wawancara dengan salah satu pemilik dari UMKM Tofan *Computer* dan dianalisis dari sesi tanya jawab yang dilakukan, kemudian dikumpulkan menjadi sebuah daftar. Tabel 3 adalah tabel daftar kebutuhan pengguna berdasarkan hasil analisis berisikan kebutuhan fungsional dari pengguna dalam pelaksanaan transaksi penjualan UMKM Tofan *Computer*.

Tabel 3. Daftar Kebutuhan Pengguna

Daftar Kebutuhan Pengguna (<i>User Requirements</i>)		
No.	Uraian	Kode
1.	Pemilik ingin adanya aplikasi untuk membantu pelaksanaan transaksi penjualan menggunakan smartphone	UR01
2.	Pemilik ingin akses untuk masuk ke aplikasi hanya untuk pengguna terdaftar	UR02
3.	Pemilik ingin menggunakan email pribadi sebagai akses masuk aplikasi	UR03
4.	Pemilik ingin mendaftarkan akun dibatasi dengan jabatan pemilik	UR04
5.	Pemilik dapat menghapus akun pengguna	UR05
6.	Pemilik dapat mendaftarkan outlet	UR06
7.	Pemilik dapat menghapus outlet	UR07
8.	Pemilik dan penjual dapat melihat daftar produk	UR08
9.	Pemilik dan penjual dapat menambahkan data produk	UR09

Lanjutan Tabel 3. Daftar Kebutuhan Pengguna

Daftar Kebutuhan Pengguna (<i>User Requirements</i>)		
No.	Uraian	Kode
10.	Pemilik dan penjual dapat mencari data produk	UR10
11.	Pemilik dan penjual dapat mengubah data produk	UR11
12.	Pemilik dan penjual dapat menghapus data produk	UR12
13.	Pemilik dan penjual dapat melakukan transaksi penjualan	UR13
14.	Pemilik dan penjual dapat membuat struk transaksi penjualan	UR14
15.	Pemilik dan penjual dapat melihat riwayat transaksi	UR15
16.	Pemilik dapat melihat laporan penjualan	UR16
17.	Pemilik dapat mencari laporan penjualan	UR17
18.	Pemilik dan penjual dapat mengubah profil akun	UR18

Sumber: Hasil Olah Data Penulis, 2023

Software requirements menjadi hal yang sangat penting pada penelitian ini. Identifikasi ini dilakukan untuk menentukan batasan dan gambaran dari aplikasi yang dibutuhkan. *Requirements* ini dibuat berdasarkan *business requirements* dan *user requirement*. Kebutuhan perangkat lunak dalam penelitian ini dibagi menjadi kebutuhan fungsional dan kebutuhan non fungsional.

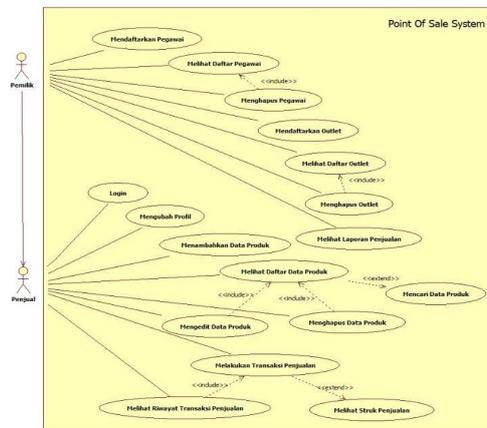
Tabel 4. *Software Requirements*

Kebutuhan Perangkat Lunak (<i>Software Requirements</i>)		
No.	Uraian	Kode
Fungsional:		
1.	Aplikasi mampu menyediakan fitur login dengan email sebagai identitasnya	SR01
2.	Aplikasi memungkinkan pemilik mendaftarkan pengguna baru	SR02
3.	Aplikasi memungkinkan pemilik menghapus data pengguna	SR03
4.	Aplikasi mampu menyediakan fitur mengubah data profil	SR04
5.	Aplikasi mampu menyediakan fitur mengubah password	SR05
6.	Aplikasi memungkinkan pemilik mendaftarkan outlet baru	SR06
7.	Aplikasi memungkinkan pemilik menghapus data outlet	SR07
8.	Aplikasi mampu menyediakan fitur melihat daftar produk	SR08
9.	Aplikasi mampu menyediakan fitur menambahkan produk	SR09
10.	Aplikasi mampu menyediakan fitur mencari data produk	SR10
11.	Aplikasi mampu menyediakan fitur mengubah data produk	SR11
12.	Aplikasi mampu menyediakan fitur menghapus data produk	SR12
13.	Aplikasi memungkinkan pengguna melakukan transaksi penjualan	SR13
14.	Aplikasi mampu membuat struk berdasarkan transaksi penjualan	SR14
15.	Aplikasi mampu menyediakan fitur melihat riwayat transaksi penjualan	SR15
16.	Aplikasi memungkinkan pemilik melihat laporan penjualan	SR16
17.	Aplikasi memungkinkan pemilik mencari laporan penjualan	SR17
Non Fungsional:		
18.	Aplikasi mampu berjalan pada perangkat smartphone dan terintegrasi antara 1 perangkat dengan perangkat lainnya	SR18
19.	Aplikasi membutuhkan akses internet agar terhubung ke layanan database server	SR19
20.	Aplikasi mampu menampilkan logo perusahaan pada halaman awal aplikasi	SR20
21.	Aplikasi mampu membatasi penggunaan email untuk satu pengguna	SR21
22.	Aplikasi mampu membagikan struk penjualan	SR22

Sumber: Hasil Olah Data Penulis, 2023

Tabel 4 menunjukkan kebutuhan yang diperlukan untuk membangun aplikasi yang dirancang. Kebutuhan fungsional tersebut adalah jenis kebutuhan yang berisi proses apa saja yang dilakukan. Sedangkan kebutuhan non fungsional adalah jenis kebutuhan yang berisi properti perilaku yang dimiliki aplikasi, meliputi operasional, kinerja, dan informasi.

3.2 Mengumpulkan *Requirements*



Sumber: Hasil Olah Data Penulis, 2023

Gambar 3. Use Case Diagram Point of Sale System

Membuat model fungsional dilakukan untuk memodelkan *requirements* berdasarkan kebutuhan fungsional yang sudah diidentifikasi. Kebutuhan fungsional tersebut dijadikan model fungsional dengan menggunakan *use case diagram*. Gambar 3 merupakan *use case diagram* dari sistem POS yang ingin dibangun. Pada diagram tersebut dapat dilihat 'Pemilik' merupakan aktor khusus yang mewarisi perilaku dari aktor yang lebih umum yaitu 'Penjual'. Pada diagram tersebut juga terdapat banyak *use case*. Selanjutnya akan menjelaskan beberapa *use case* secara mendetail dengan membuat *use case description*-nya.

3.2.1 *Use Case Description* Mendaftarkan Pegawai

Pemilik perlu mendaftarkan akun pegawai (UR04), dijadikan persyaratan tersebut menjadi sebuah kebutuhan fungsional *software* (SR02). Kebutuhan fungsional tersebut dimodelkan menjadi seperti berikut.

Berdasarkan Tabel 5 ditunjukkan bahwa *use case* tersebut memiliki prioritas tinggi yang artinya penting atau menjadi aktivitas utama yang harus dilakukan. Untuk memulai *use case* tersebut pemilik harus login terlebih dahulu ke dalam aplikasi. Pada alur proses ketiga, terdapat opsi tambahan jika pemilik ingin menambahkan foto awal maupun tidak. Dan terdapat beberapa kondisi alternatif yang dipakai diantaranya yaitu jika format *email* yang diisi salah, jika konfirmasi *password* yang diisi tidak sama dengan *password* yang diisi sebelumnya dan juga jika terdapat bagian kosong atau belum diisi maka alur proses tidak bisa dilanjutkan.

Tabel 5. *Use Case Description* Mendaftarkan Pegawai

Use Case Name : Mendaftarkan Pegawai	ID: UC-02	Priority : High
Actor : Pemilik		
Description : Use case ini menjelaskan bagaimana pemilik mendaftarkan pegawai		
Trigger : Pemilik ingin mendaftarkan akun pegawai lainnya		
Preconditions : Pemilik sedang login ke dalam aplikasi		
Normal Flow of Events : <ol style="list-style-type: none"> 1. Pemilik menekan menu Pegawai 2. Pemilik menekan tombol tambah pegawai 3. Pemilik memasukkan data awal pegawai Jika pemilik ingin menambahkan photo awal pegawai, Sub-Flow S-1 dijalankan. Jika pemilik tidak ingin menambahkan photo awal pegawai, Sub-Flow S-2 dijalankan. <ol style="list-style-type: none"> 4. Pemilik mengisi nama pegawai 5. Pemilik mengisi email pegawai 6. Pemilik mengisi password pegawai 7. Pemilik mengisi konfirmasi password pegawai 8. Pemilik menekan tombol Konfirmasi 9. Pemilik kembali ke menu Pegawai 		
SubFlows : S-1 : Pemilik ingin menambahkan photo awal pegawai <ol style="list-style-type: none"> 1. Pemilik menekan icon plus pada field photo 2. Pemilik memilih photo dari galery 3. Proses dilanjutkan ke nomor 4 S-2 : Pemilik tidak ingin menambahkan photo awal pegawai <ol style="list-style-type: none"> 1. Photo dibiarkan menggunakan photo default 2. Proses dilanjutkan ke nomor 4 		
Alternate/Exceptional Flows : E-1 : Format email yang dimasukkan salah 5e.1 Muncul pesan kesalahan format email salah 5e.2 Pemilik harus mengisikan email kembali E-2 : Konfirmasi password yang dimasukkan tidak sama dengan password 7e.1 Muncul pesan kesalahan password tidak sama 7e.2 Pemilik harus mengisikan konfirmasi password kembali E-3 : field email atau password belum diisi atau kosong 8e.1 Tombol konfirmasi tidak bisa ditekan 8e.2 Proses tidak bisa dilanjutkan		

Sumber: Hasil Olah Data Penulis, 2023

3.2.2 *Use Case Description* Melakukan Transaksi

Pemilik dan penjual perlu melakukan transaksi penjualan (UR13), dijadikan persyaratan tersebut menjadi sebuah kebutuhan fungsional *software* (SR13). Kebutuhan fungsional tersebut dimodelkan menjadi seperti berikut.

Tabel 6. *Use Case Description* Melakukan Transaksi

Use Case Name : Melakukan Transaksi	ID: UC-14	Priority : High
Actor : Penjual		
Description : Use case ini menjelaskan bagaimana penjual melakukan transaksi penjualan		
Trigger : Penjual ingin menjual produk yang tersedia		
Preconditions : Penjual sedang login ke dalam aplikasi		
Normal Flow of Events : <ol style="list-style-type: none"> 1. Penjual menekan menu Transaksi 2. Penjual melihat daftar data produk Jika penjual ingin mencari produk tertentu, Sub-Flow S-1 dijalankan. <ol style="list-style-type: none"> 3. Penjual memilih produk-produk yang akan dijual 4. Penjual menekan produk sesuai dengan banyak produk yang ingin dijual 5. Sistem mengkalkulasikan jumlah produk dan jumlah harga produk yang dipilih 6. Penjual menekan tombol Tagih 7. Penjual memilih outlet transaksi dilakukan 8. Penjual menekan tombol checkout 9. Penjual melihat struk transaksi 		

Sumber: Hasil Olah Data Penulis, 2023

Berdasarkan Tabel 6, *use case* melakukan transaksi memiliki prioritas tinggi yang artinya penting untuk dilakukan dan untuk memulai *use case* tersebut juga harus *login* terlebih dahulu ke dalam aplikasi. Sebelumnya dijelaskan bahwa “Pemilik” dan “Penjual” perlu melakukan transaksi

namun di *use case description* diatas hanya disebutkan “Penjual” saja. Hal itu dikarenakan pada Gambar 3 *Use Case Diagram* dijelaskan bahwa pemilik mewarisi perilaku yang dimiliki “Penjual”.

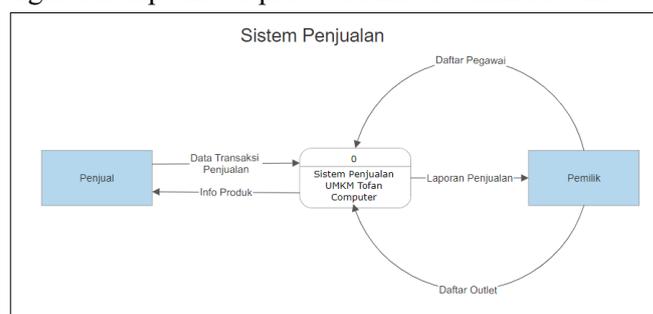
Tabel 7. *Use Case Description* Melakukan Transaksi Lanjutan

<p>SubFlows :</p> <p>S-1 : Penjual ingin mencari produk</p> <ol style="list-style-type: none"> 1. Penjual mengisi field pencarian berdasarkan nama produk atau merk produk 2. Data produk yang sesuai dengan pencarian ditampilkan 3. Proses dilanjutkan ke nomor 3
<p>Alternate/Exceptional Flows :</p> <p>E-1 : Jumlah produk yang dipilih habis atau tidak cukup</p> <ol style="list-style-type: none"> 4e.1 Muncul pop up stok produk kurang 4e.2 Penjualan produk tersebut tidak bisa dilanjutkan <p>E-2 : Tidak ada outlet yang dipilih</p> <ol style="list-style-type: none"> 7e.1 Muncul pop up outlet belum dipilih 7e.2 Proses tidak bisa dilanjutkan

Sumber: Hasil Olah Data Penulis, 2023

Pada alur proses kedua (Tabel 7), *use case* tersebut menggambarkan peristiwa ketika sedang melihat daftar produk. “Penjual” dapat mencari produk tertentu yang dirasa sulit ditemukan karena banyaknya data produk dalam daftar. Terdapat kondisi alternatif dimana ketika jumlah produk yang dipilih habis ataupun memiliki stok yang kurang, penjual tidak dapat melanjutkan proses penjualan dengan produk tersebut. Kondisi ini juga memberikan pengecualian ketika penjual lupa atau belum memilih *outlet* tempat penjual bertransaksi maka proses transaksi tidak dapat dilanjutkan.

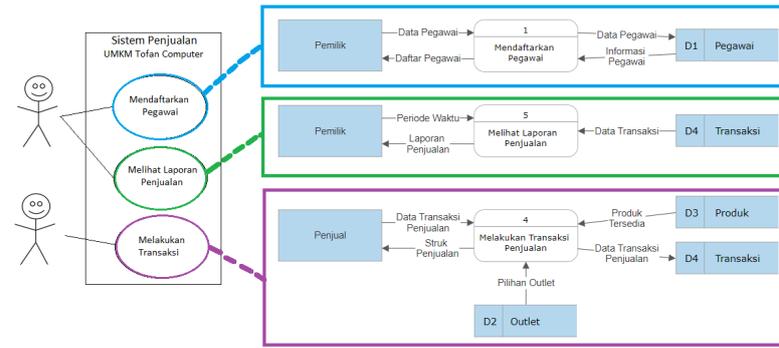
Model proses dalam pengembangan sistem aplikasi POS dibuat untuk menggambarkan aktivitas-aktivitas yang dilakukan dan bagaimana perpindahan data dalam aktivitas tersebut. Dalam penulisan ini, menggunakan teknik *Data Flow Diagram* (DFD) untuk membuat model prosesnya. Proses bisnis terlalu kompleks jika digambarkan dengan satu DFD, sehingga dilakukan dekomposisi untuk mengurai kompleksitas proses bisnis di dalam sistem.



Sumber: Hasil Olah Data Penulis, 2023

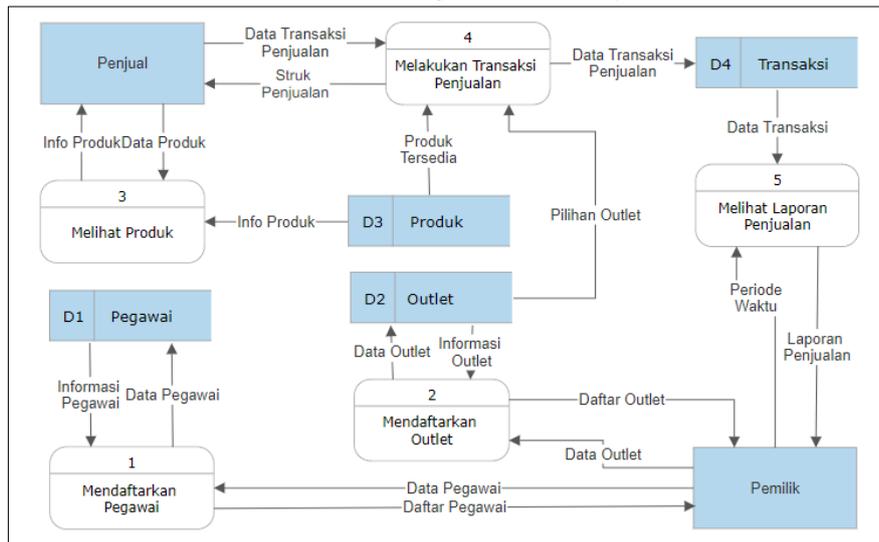
Gambar 4. Diagram Konteks Sistem Penjualan

Gambar 4 merupakan DFD pertama pada proses bisnis yang memperlihatkan proses keseluruhan hanya sebagai satu proses saja. Pada diagram konteks tersebut, terdapat dua *external entity* yaitu *actor* yang disebutkan dalam model sebelumnya. Keduanya mendapatkan dan memberikan informasi dari dan ke dalam sistem.



Sumber: Hasil Olah Data Penulis, 2023

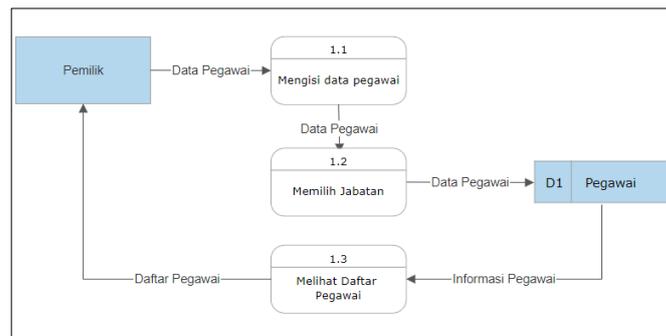
Gambar 5. DFD Fragments Sistem Penjualan



Sumber: Hasil Olah Data Penulis, 2023

Gambar 6. Level 0 Diagram Sistem Penjualan

DFD *fragments* disusun berdasarkan *use case - use case* sebelumnya. Setiap *use case* dikonversi menjadi sebuah DFD *fragment*. Penelitian ini hanya melampirkan tiga *use case* saja sesuai dengan pembuatan model sebelumnya. Jumlah proses pada DFD sama dengan jumlah *use case*. Penyusunan ini dilakukan, agar memudahkan dalam membuat diagram level 0 seperti pada Gambar 6. Diagram Level 0 menampilkan semua proses utama dari keseluruhan sistem dan menunjukkan bagaimana proses-proses tersebut saling terkait oleh aliran data. Proses dan data yang lebih detail dijelaskan pada DFD Level 1 (Gambar 7), sebagai contoh pada proses mendaftarkan pegawai. Pada UMKM Tofan *Computer* proses mendaftarkan pegawai baru dilakukan oleh pemilik UMKM. Prosesnya membutuhkan data pegawai untuk didaftarkan ke dalam sistem. Pemilik mengisi dan memilih jabatan pegawai. Data pegawai yang sudah diisi akan disimpan ke dalam basis data. Pemilik dapat mengetahui daftar pegawai yang ada dengan melihat informasi pegawai pada basis data tersebut.



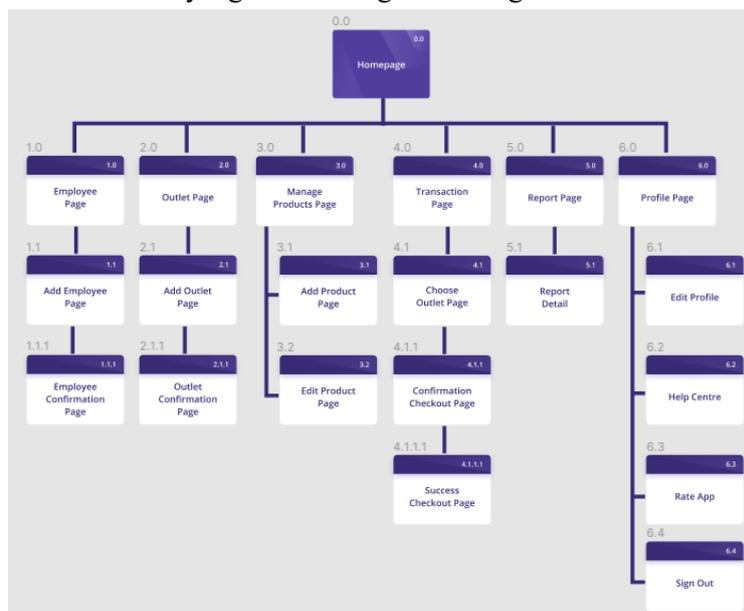
Sumber: Hasil Penelitian, 2023

Gambar 7. Level 1 Diagram Proses Mendaftarkan Pegawai

Selanjutnya disusun model data untuk menentukan dan menganalisis persyaratan data yang diperlukan untuk mendukung proses bisnis. Pada penelitian ini digunakan Cloud firestore. Cloud firestore adalah database NoSQL yang berorientasi dokumen. Tidak seperti *database SQL*, database ini tidak memiliki tabel ataupun baris. Sebagai gantinya menyimpan data dalam *document* yang disusun menjadi *collection*. Semua *document* harus disimpan dalam *collection*. *Document* dapat berisi *subcollections* dan *nested objects*, keduanya dapat menyertakan *primitive fields* seperti *string* atau *complex objects* seperti *list*. Penelitian menggunakan 4 buah *collections* pada penelitian ini, salah satunya adalah transaksi. *Collection* transaksi menyimpan *document* dari transaksi. *Document* ini diberikan nama unik dengan nama identitas transaksi tersebut dan setiap *document* berisikan sekumpulan pasangan *key-value*.

3.3 Membuat Rancangan

Langkah pertama membuat rancangan adalah merancang arsitektur informasi. Merancang arsitektur informasi berguna sebagai penuntun rancangan sistem yang dibuat dan menjadi cetak-biru untuk segala arahan di masa yang akan datang atau sebagai dokumentasi dari sistem.

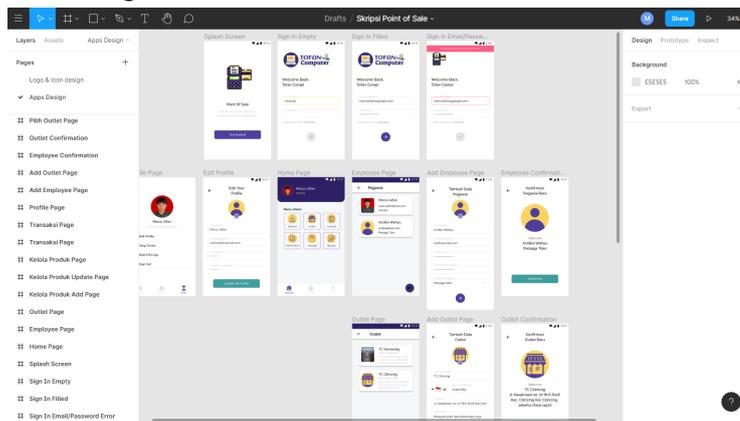


Sumber: Hasil Olah Data Penulis, 2023

Gambar 8. Sitemap System

Gambar 8 merupakan peta situs (*sitemap*) sistem dari arsitektur informasi yang dibuat. Peta tersebut menjelaskan halaman-halaman apa saja yang ada pada sistem dan bagaimana hubungan antara halaman tersebut. Sebagai contoh agar sistem dapat mengakses halaman konfirmasi pegawai dari beranda, sistem perlu melewati 2 halaman sebelumnya yaitu halaman pegawai dan halaman tambah data pegawai.

Langkah kedua adalah merancang antarmuka (*interface*) yang berguna untuk memahami kebutuhan pengguna dan menerjemahkannya menjadi elemen-elemen desain yang memadai, sehingga menghasilkan antarmuka yang sesuai dengan harapan pengguna. Sebelum rancangan yang dibuat menjadi sebuah prototipe sistem yang dapat disajikan perlu merancang antarmuka aplikasi yang dibangun. Dalam perancangan ini dibuat *mockup* antarmuka halaman-halaman yang ada pada aplikasi dengan bantuan *tools* Figma.

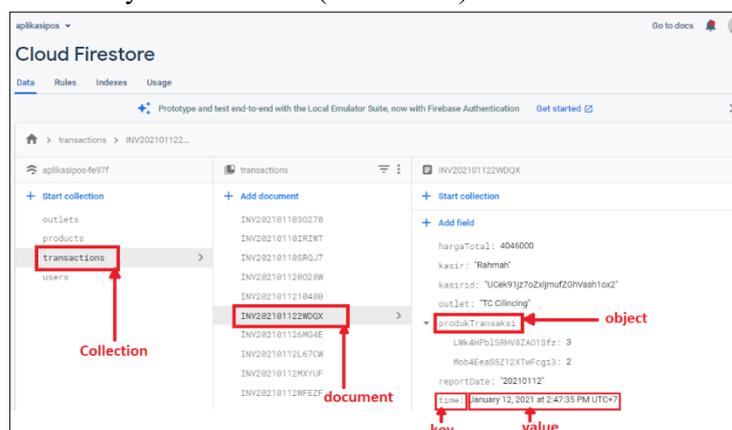


Sumber: Hasil Olah Data Penulis, 2023

Gambar 9. Draft Project Figma

Gambar 9 merupakan halaman kerja atau *draft project* Figma yang dilakukan, terlihat terdapat beberapa *mockup* antarmuka dari halaman-halaman aplikasi. Selanjutnya diperlihatkan secara lebih jelas dengan membaginya berdasarkan skenario model yang telah dibuat sebelumnya.

Selanjutnya perancangan basis data dalam penelitian ini menggunakan *cloud* Firestore sebagai *database*. Dalam penerapannya, aplikasi harus terhubung ke internet untuk terkoneksi atau menyimpan data ke dalam layanan *database* (Gambar 10).



Sumber: Hasil Olah Data Penulis, 2023

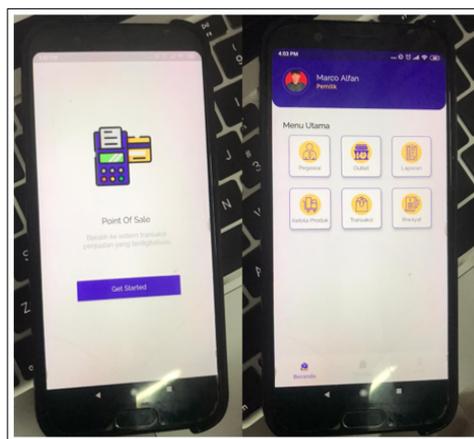
Gambar 10. Console Cloud Firestore Project

Gambar 10 merupakan *console* dari *cloud* Firestore aplikasi yang dibangun. Gambar tersebut menampilkan rancangan basis data yang digunakan berdasarkan model data yang dibuat

sebelumnya dengan menggunakan 4 *collection*. Contoh pada *collection* ‘*transactions*’ mempunyai beberapa *document* di dalamnya. *Document* tersebut merupakan identitas dari data-data transaksi yang sudah dilakukan. Setiap *document* tersebut mempunyai *field* yang berisikan rincian data dari transaksi tersebut. Setiap *document* yang berada pada *collection* ‘*transactions*’, memiliki *field* dan tipe data yang sama antara lain ‘*hargaTotal*’, ‘*kasir*’, ‘*kasirId*’, ‘*outlet*’, ‘*produkTransaksi*’, ‘*reportDate*’, ‘*time*’.

3.4 Mengembangkan Purwarupa

Setelah langkah-langkah pengumpulan *requirements* hingga perancangan telah dilakukan, selanjutnya dilakukan pengembangan purwarupa. Pada penelitian ini pengembangan dilakukan dengan metode *coding* menggunakan bahasa pemrograman Dart dengan menggunakan *Framework* Flutter. Penelitian ini menggunakan Visual Studio Code Versi 1.47.0 sebagai *text editor* yang berjalan pada sistem operasi Windows 10.



Sumber: Hasil Olah Data Penulis, 2023

Gambar 11. Hasil Purwarupa Aplikasi

Gambar 11 merupakan hasil purwarupa yang sudah diimplementasikan pada perangkat Android. Pada halaman ‘*splash screen*’ diberikan logika proses di dalamnya. Ketika belum ada pengguna yang login ke dalam aplikasi, maka akan terlihat tampilan seperti Gambar 11 bagian kiri, kemudian pengguna akan melalui proses login. Sedangkan ketika pengguna sudah atau masih login ke dalam aplikasi, maka ketika membuka aplikasi kembali akan terlihat tampilan seperti Gambar 11 bagian kanan yaitu halaman beranda yang berisi tampilan menu utama.

3.5 Menguji Purwarupa

Proses pengujian purwarupa dilakukan dengan metode unit *testing* dengan teknik *blackbox testing*, berfokus kepada fungsional dari purwarupa dengan meninjau masukan dan keluarannya. Hal ini dilakukan untuk memastikan setiap unit kode dari purwarupa dapat bekerja sesuai harapan.

Pengujian pertama dilakukan pada *case* mendaftarkan pegawai, dengan hasil seperti pada Tabel 8, Hasil pengujian ini memberikan informasi bahwa pengembangan purwarupa untuk *case* mendaftarkan pegawai berhasil dilakukan. Pengembangan aplikasi POS untuk *case* mendaftarkan pegawai merupakan salah satu bagian dari model fungsional yang sebelumnya sudah dibuat (UC02), model tersebut juga sesuai dengan persyaratan yang telah diminta oleh pengguna atau permintaan stakeholder merujuk pada (UR02).

Tabel 8. *Test Case* Mendaftarkan Pegawai

Test Case Name: Mendaftarkan Pegawai	ID: TC01	
Skenario	Hasil yang diharapkan	Status
Pemilik menekan menu pegawai pada bagian menu utama halaman beranda	Masuk ke halaman pegawai	berhasil
Pemilik menekan tombol '+' pada halaman pegawai	Masuk ke halaman tambah data pegawai	berhasil
Pemilik menambahkan foto awal pegawai	Gambar <i>icon default</i> berganti dengan foto yang dipilih	berhasil
Pemilik mengisi formulir pendaftaran pegawai dengan <i>email</i> dan <i>password</i>	Bidang formulir berisi masukan yang telah di isi.	berhasil
Pemilik memilih jabatan awal pegawai pada <i>combo box</i>	<i>Combo box</i> menampilkan jabatan	berhasil
Pemilik menekan tombol selanjutnya	Masuk ke halaman konfirmasi pegawai dengan isian formulir	berhasil
Pemilik menekan tombol konfirmasi	Data pegawai disimpan ke dalam <i>database</i> , dan ke halaman pegawai	berhasil
Database Terkait: 'users'		
Catatan:		

Sumber: Hasil Olah Data Penulis, 2023

Tabel 9. *Test case* melakukan transaksi

Test Case Name: Melakukan Transaksi	ID: TC02	
Skenario	Hasil yang diharapkan	Status
Penjual menekan menu transaksi pada bagian menu utama halaman beranda	Masuk ke halaman transaksi	berhasil
Penjual memilih produk-produk yang dingin dijual	Produk yang dipilih masuk ke dalam keranjang	berhasil
Penjual menekan produk sesuai dengan banyak produk yang ingin di jual	Stok produk berkurang setiap ditekan, muncul notifikasi apabila stok produk sudah tidak cukup	berhasil
Sistem mengkalkulasikan harga total produk yang dipilih	Nominal pada tombol tagih berubah sesuai harga total	berhasil
Penjual menekan tombol tagih	Beralih ke halaman pilih outlet	berhasil
Penjual memilih outlet dan menekan tombol selanjutnya	Beralih ke halaman konfirmasi checkout dengan isian produk dan outlet yang dipilih	berhasil
Penjual menekan tombol konfirmasi	Data transaksi disimpan ke dalam <i>database</i> , dan ke halaman sukses transaksi	berhasil
Penjual menekan tombol lihat struk	Menampilkan struk penjualan	berhasil
Database Terkait: 'products', 'outlets', 'transactions'		
Catatan: Fitur bagikan struk penjualan hanya menyimpan struk ke dalam galeri penyimpanan		

Sumber: Hasil Olah Data Penulis, 2023

Nilai lebih dari pengembangan aplikasi POS menggunakan *Framework* Flutter dengan metode RAD pada case ini adalah peningkatan efisiensi dan akurasi dalam proses pendaftaran pegawai sesuai dengan permintaan *stakeholder* dengan menempatkan semua identifikasi kebutuhan di awal. Dengan adanya aplikasi POS yang telah dikembangkan dengan metode RAD, proses pendaftaran pegawai dapat dilakukan secara cepat, ter otomatisasi, dan mengurangi potensi kesalahan manusia.

4. PENUTUP

Pengembangan aplikasi *point of sale* pada UMKM Tofan *Computer* memiliki 22 kebutuhan perangkat lunak yang dibagi menjadi klasifikasi fungsional dan klasifikasi non-fungsional. Kebutuhan tersebut didapat berdasarkan hasil analisis dari kebutuhan bisnis dan kebutuhan pengguna dari UMKM Tofan *Computer*. Daftar kebutuhan perangkat lunak dapat dimodelkan menjadi model fungsional, model proses dan model data. Dalam merancang aplikasi *Point of Sale* (POS) berbasis *mobile* menggunakan *Framework* Flutter terdapat beberapa metode perancangan yang dibuat berdasarkan dari kebutuhan perangkat lunak yang telah dimodelkan. Metode pertama yaitu perancangan arsitektur informasi yang menghasilkan *sitemap* dari sistem. Metode kedua perancangan antarmuka yang menghasilkan *mockup* dari setiap halaman aplikasi. Metode ketiga yaitu perancangan basis data, pada metode ini menunjukkan *database* hingga tipe data seperti apa yang digunakan. Rancangan tersebut dikembangkan hingga menjadi purwarupa. Purwarupa yang dihasilkan dapat berjalan pada perangkat android dan telah melalui proses *unit testing* dengan teknik pengujian *black box*. Pemilik UMKM Tofan *Computer* dapat mengetahui stok produk, melakukan transaksi hingga melihat laporan penjualan dengan mudah melalui aplikasi *point of sale* yang dibuat.

Berdasarkan hasil analisis dan perancangan aplikasi *Point of Sale* (POS) menggunakan *Framework* Flutter dengan metode *Rapid Application Development* (RAD), terdapat beberapa saran yang dapat diberikan. Pertama, metode RAD cocok digunakan untuk pengembangan perangkat lunak yang memiliki kebutuhan yang dapat berubah dengan cepat dan membutuhkan waktu pengembangan yang relatif singkat. Metode ini fokus pada kecepatan dan fleksibilitas dalam menghasilkan produk perangkat lunak yang dapat segera digunakan oleh pengguna. Kedua, UMKM lain dapat mempertimbangkan untuk mengembangkan aplikasi POS-nya sendiri dengan mengacu pada penelitian ini sebagai referensi. Dengan mengikuti saran-saran tersebut, diharapkan UMKM dapat meningkatkan efisiensi dan produktivitas dalam proses penjualan sehingga dapat bersaing lebih baik di pasar. Selain itu, aplikasi POS yang terintegrasi dan handal juga dapat memberikan pengalaman berbelanja yang lebih baik bagi pelanggan.

Ucapan Terimakasih

Penulis menyampaikan ucapan terima kasih kepada pihak-pihak yang telah berkontribusi dalam penelitian. Terutama kepada Bapak Virgianto Rachmawan selaku *stakeholder* dan semua pihak UMKM Tofan *Computer* yang telah mengizinkan saya untuk melakukan penelitian di tempat usahanya tersebut dan kepada Bapak Dr. Berliyanto, M.TI selaku pembimbing yang telah berkenan memberikan motivasi, dan arahan serta meluangkan banyak waktu guna membimbing dalam penulisan untuk selesainya penyusunan penelitian ini.

DAFTAR PUSTAKA

- Ahyar, H., Maret, U. S., Andriani, H., Sukmana, D. J., Mada, U. G., Hardani, S.Pd., M. S., Nur Hikmatul Auliya, G. C. B., Helmina Andriani, M. S., Fardani, R. A., Ustiawaty, J., Utami, E. F., Sukmana, D. J., & Istiqomah, R. R. (2020). *Buku Metode Penelitian Kualitatif & Kuantitatif* (Issue March).
- Ardy, U., Desiana, J., & Rachmadi, M. (2018). Pengembangan aplikasi perencanaan wisata “plesir” berbasis

- android dan ios. *Ijccs*, *x(x)*, 1–11.
- Astuti, F., & Hermawan, Y. (2020). Pemanfaatan Firebase Realtime Database Pada Aplikasi Pembelajaran Agama Islam Menggunakan *Framework Flutter*. *Issn 2715-2944*, *18(2)*, 16.
- Chandra, K. (2017). Rancang Bangun Sistem Informasi Point of Sales Berbasis Website Pada Distributor Kain Hoggy Djaya. *Sistem Informasi*, *1(3)*, 192–200.
- BPS - Badan Pusat Statistik. (2020). Laporan Bulanan UMKM Indonesia Tahun 2020. <https://www.bps.go.id/publication/2020/10/30/ae2482c5b1a663e6081feacc/laporan-bulanan-umkm-indonesia-terkini-oktober-2020.html>
- Dennis, A. (2012). *Systems Analysis and Design, 5th Edition*. John Wiley & Sons. <https://books.google.co.id/books?id=QfQbAAAAQBAJ>
- Dennis, Alan, Wixom, B. H., & Tegarden, D. (2015). System Analysis and Design: An object-oriented approach with UML, 5th ed. In *Journal of Chemical Information and Modeling* (Vol. 53, Issue 9).
- Haqi, B., Kom, M., & Setiawan, H. S. (2019). *Aplikasi Absensi Dosen Dengan Java dan Smartphone Sebagai Barcode Reader*. Elex Media Komputindo.
- Iswari, N. M. S. (2015). Review Perangkat Lunak StarUML Berdasarkan Faktor Kualitas McCall. *Ultimatics*, *VII(1)*, 72–81.
- Kementerian Koperasi dan UKM. (2018). Perkembangan Data Usaha Mikro , Kecil , Menengah Dan Usaha Besar. *Www.Depkop.Go.Id*, *2000(1)*, 1.
- Napoli, M. (2020). Beginning Flutter: A Hands On Guide To App Development. In *Allitebooks*. <https://doi.org/10.1002/9781119550860>
- Nuryantono, N., & Kristiani, E. (2019). Faktor-Faktor Yang Mempengaruhi Keputusan Pengadopsian Sistem Point Of Sale (POS) Pada Usaha Mikro Kecil Menengah (UMKM) di Kota Makassar. *Jurnal Ilmiah Teknik Industri*, *17(1)*, 43-50.
- Pamungkas, G., & Yuliansyah, H. (2017). Rancang Bangun Aplikasi Android Pos (Point of Sale) Kafe Untuk Kasir Portable Dan Bluetooth Printer. *JST (Jurnal Sains Dan Teknologi)*, *6(1)*, 199–208. <https://doi.org/10.23887/jst-undiksha.v6i1.8828>
- Putra, B. A. (2018). *Rancang Bangun Aplikasi Point of Sale Berbasis Web Responsif untuk Ponsel Pintar*. Institut Teknologi Sepuluh Nopember.
- Setiyani, L. (2019). [*Software Engineering*] Lila Setiyani , S. T, M. Kom. May, 20–25.
- Sherly Siregar, & Adinda Rizki Mulyani. (2021). Pemanfaatan Aplikasi Point Of Sale (Pos) Terhadap Efisiensi Penjualan Pada Usaha Mikro Kecil Menengah (Umkm) Di Kecamatan Sungai Geringging Kabupaten Solok. *Prosiding Seminar Nasional Ilmu Ekonomi*, *8(2)*, 98-108.
- Sugihartono, J., Satoto, K. I., & Widiyanto, E. D. (2015). Pembuatan Aplikasi Point of Sale Toko Cabang Perusahaan Torani Menggunakan Framework CodeIgniter. *Jurnal Teknologi Dan Sistem Komputer*, *3(4)*, 445. <https://doi.org/10.14710/jtsiskom.3.4.2015.445-455>
- Sukamto, R. A., & Shalahuddin, M. (2016). *Rekayasa perangkat lunak terstruktur dan berorientasi objek*.
- Syafaruddin, S., & Hamidah, N. (2020). Analisis pengelolaan transaksi penjualan pada UMKM di era digital. *Jurnal Rekayasa*, *14(2)*, 117-125.
- We Are Social & Hootsuite. (2020). Indonesia Digital report 2020. *Global Digital Insights*, 1–92. <https://datareportal.com/reports/digital-2020-global-digital-overview>

