

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu bertujuan untuk mendapatkan bahan perbandingan dan acuan. Selain itu, untuk menghindari anggapan kesamaan dengan penelitian ini. Maka dalam kajian pustaka ini peneliti mencantumkan hasil-hasil penelitian terdahulu sebagai berikut:

1. Penelitian oleh Agit Amrullah dan Ema Utami dengan judul “Perancangan Sistem Informasi Pada Smart UMKM dalam Mendukung Sleman Smart Regency”. Penelitian ini merupakan penelitian yang menggunakan metode TOGAF *Architecture Development Method* (ADM). Hasil penelitian yaitu perancangan aplikasi Smart UMKM didapatkan 5 bentuk sistem informasi yaitu Perijinan UMKM, Pembinaan UMKM, Pemasaran UMKM, Aspirasi UMKM, dan Pemantauan UMKM. (Amrullah dan Utami, 2018)
2. Penelitian oleh Issa Dyah Utami, Iwan Santosa, Rulli Annisa, Irda Rebecca Simanjuntak, Ulfi Atika Permatasari, Nur Faizah dengan judul “Sistem Informasi Manajemen Berbasis Web pada Usaha Mikro Kecil Dan Menengah (UMKM) (Studi Kasus: UMKM Sumber Jokotole, Socah, Bangkalan)”. Penelitian ini merupakan penelitian yang menggunakan metode objek orientasi. Hasil penelitian yaitu Sistem informasi akan memberikan manfaat bagi produsen, supplier maupun konsumen untuk mengatur proses bisnis pada UMKM kerupuk. (Utami *et al.*, 2019)
3. Penelitian oleh Yonanda Aditama Sari, Ridho Pamungkas, Andria dengan judul “Rancang Bangun Sistem Informasi Sebagai Media Promosi Untuk Usaha Mikro Kecil Menengah (UMKM)”. Penelitian ini merupakan penelitian yang menggunakan metode waterfall. Hasil penelitian yaitu Website UMKM Bersinar cukup memadai dan mudah digunakan bagi masyarakat Desa Goranggareng untuk memajukan UMKM Desa tersebut. (Sari *et l.*, 2022)

Tabel 2.1 Penelitian Terdahulu

Nama	Judul	Variabel	Metode Analisis	Hasil Analisis
(Amrullah and Utami, 2018)	Perancangan Sistem Informasi Pada Smart UMKM dalam Mendukung Sleman Smart Regency	UMKM, Smart Regency, TOGAF ADM, Sistem Informasi, Smart City	TOGAF <i>Architecture Development Method</i> (ADM)	Hasil penelitian yaitu perancangan aplikasi Smart UMKM didapatkan 5 bentuk sistem informasi yaitu Perijinan UMKM, Pembinaan UMKM, Pemasaran UMKM, Aspirasi UMKM, dan Pemantauan UMKM
(Utami <i>et al.</i> , 2019)	Sistem Informasi Manajemen Berbasis Web pada Usaha Mikro Kecil Dan Menengah (UMKM) (Studi Kasus: UMKM Sumber Jokotole, Socah, Bangkalan)	Sistem informasi manajemen, Usaha mikro kecil dan menengah, pemasaran	Objek Orientasi	Sistem informasi akan memberikan manfaat bagi produsen, supplier maupun konsumen untuk mengatur proses bisnis pada UMKM kerupuk.
(Sari <i>et al.</i> , 2022)	Rancang Bangun Sistem Informasi Sebagai Media Promosi Untuk Usaha Mikro Kecil Menengah (UMKM)	UMKM Bersinar, Manejemen, sistem informasi	Waterfall	Website UMKM Bersinar cukup memadai dan mudah digunakan bagi masyarakat Desa Goranggareng untuk memajukan UMKM Desa tersebut.

Berdasarkan penelitian tersebut maka perbedaan pada metode yaitu peneliti menggunakan metode *extreme programming* sebagai pengembangan sistem dan pada sub sistem peneliti menambahkan permohonan bantuan yang diberikan kepada pelaku umkm seperti koperasi.

2.2 Sistem Informasi

Sistem Informasi adalah sistem yang terdiri dari orang-orang dan komputer yang memproses atau menafsirkan informasi. Sistem informasi ini menyangkut *hardware*, *software*, infrastruktur dan orang-orang. Sistem merupakan Sekumpulan unsur/ elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Sistem juga terdiri dari beberapa komponen yang saling berhubungan dan bekerjasama untuk mencapai sasaran atau tujuan, sebagai contoh yaitu sistem pembelajaran terdiri dari guru, siswa, tujuan, metode, materi hingga penilaian (Wijoyo *et al.*, 2021).

2.3 CodeIgniter

CodeIgniter adalah sebuah *framework* php yang bersifat *open source* dan menggunakan metode MVC (*Model, View, Controller*) untuk memudahkan *developer* atau programmer. Dibandingkan *web framework* yang lain *CodeIgniter* memiliki desain yang lebih sederhana dan bersifat tidak kaku. Sebuah *framework* php yang bersifat gratis dan berukuran kecil merupakan keunggulan tersendiri jika dibandingkan framework lain (Supardi dan Hermawan, 2019).

2.4 Website

Website merupakan halaman yang menampilkan informasi data teks, gambar, suara, video atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis. Halaman pertama sebuah *website* disebut *homepage*. *Website/Laman* merupakan kumpulan informasi atau kumpulan halaman/*page* yang bisa diakses lewat jalur *internet*. Setiap orang di berbagai tempat dan segala waktu bisa menggunakannya selama terhubung secara *online* (Harianto, *et al.*, 2019).

2.5 PHP

PHP (*Hypertext Pre-processor*) adalah pemrograman (interpreter) yang melakukan proses penerjemahan baris sumber menjadi kode mesin yang dimengerti oleh komputer secara dinamis. Pengertian PHP juga merupakan singkatan dari *Hypertext Preprocessor* dengan Bahasa yang berbentuk skrip yang bersifat *server side* yang dimana proses pengerjaan kode program dilakukan di *server*, dan hasilnya akan ditampilkan di *browser* (Sihombing, 2022).

2.6 MySql

MySQL adalah sebuah *database management system* (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. *Database management system* (DBMS) MySQL multi pengguna dan bersifat gratis. Mysql digunakan sebagai wadah dalam mengelola data yang dapat disimpan didigunakan kembali dengan cara yang lebih efisien (Setyawan and Pratiwi, 2019).

2.7 Pengertian UMKM

Usaha mikro adalah Usaha produktif milik orang perorangan dan/atau badan usaha perorangan yang memiliki kriteria usaha mikro sebagaimana diatur dalam UU tersebut. Usaha kecil adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha yang bukan merupakan anak perusahaan atau bukan anak cabang yang dimiliki, dikuasai atau menjadi bagian, baik langsung maupun tidak langsung, dari usaha menengah atau usaha besar yang memenuhi kriteria usaha kecil sebagaimana dimaksud dalam UU tersebut, sedangkan usaha mikro adalah usaha ekonomi produktif yang berdiri sendiri yang dilakukan oleh perorangan atau badan usaha yang bukan merupakan anak perusahaan atau bukan cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung, dari usaha mikro, usaha kecil atau usaha besar yang memenuhi kriteria usaha mikro (UU No. 20, 2008).

Berdasarkan Undang-Undang Republik Indonesia No.20 Tahun 2008 tentang UMKM tersebut maka usaha mikro maupun usaha makro merupakan usaha produktif yang mampu memberikan kontribusi besar bagi perkembangan

perekonomian Indonesia sebab menjadi bagian peningkatan mutu dan ekspor produk yang diproduksi.

2.7.1 Ciri dan Karakteristik UMKM

Ciri-ciri UMKM dapat dijelaskan berdasarkan kelompok usahanya (Tanjung, 2017).

Berdasarkan ciri dan karakteristik UMKM sebagai berikut :

1. Usaha Mikro
 - a. Belum melakukan manajemen/pencatatan keuangan
 - b. SDM atau pengusahanya rata-rata berpendidikan sangat rendah
 - c. Tidak mengenal perbankan
 - d. Tidak memiliki izin usaha
 - e. Memiliki tenaga kerja kurang dari 4 orang
 - f. Perputaran usaha umumnya cepat
 - g. Pada umumnya pelaku usaha mikro memiliki sifat sederhana, tekun serta dapat menerima bimbingan.

2. Usaha Kecil
 - a. Sudah melakukan manajemen atau pembukuan
 - b. SDM-nya rata-rata lulusan min SMA
 - c. Sudah memiliki izin usaha
 - d. Sudah mengenal perbankan
 - e. Karyawan lebih dari 5 orang

3. Usaha Menengah
 - a. Pada umumnya telah memiliki manajemen dan organisasi yang lebih baik, teratur, dan modern
 - b. Telah melakukan manajemen keuangan dengan penerapan akuntansi
 - c. Telah melakukan pengaturan dan menjadi anggota organisasi
 - d. Kualitas SDM meningkat seperti sarjana

2.7.2 Kriteria UMKM

Kriteria UMKM dapat dikelompokkan berdasarkan jumlah aset dan omzet yang dimiliki masing-masing badan usaha yang dirumuskan pada UU nomor 20 Tahun 2008 serta pengelompokan berdasarkan jumlah karyawan (Tanjung, 2017).

Tabel 2.2 UMKM Berdasarkan Aset dan Omzet (Tanjung, 2017)

No.	Uraian	Aset	Omzet
1	Usaha Mikro	Minimum Rp 50 Juta	Maksimum Rp.300 Juta
2	Usaha Kecil	> Rp. 50 Juta – Rp. 500Juta	> Rp. 300 Juta – Rp. 2.5 Miliar
3	Usaha Menengah	> Rp. 500 Juta - < Rp.1 Miliar	Rp. 2,5 Miliar – Rp. 50 Miliar

Tabel 2.3 Kriteria UMKM Berdasarkan Jumlah Karyawan (Tanjung, 2017)

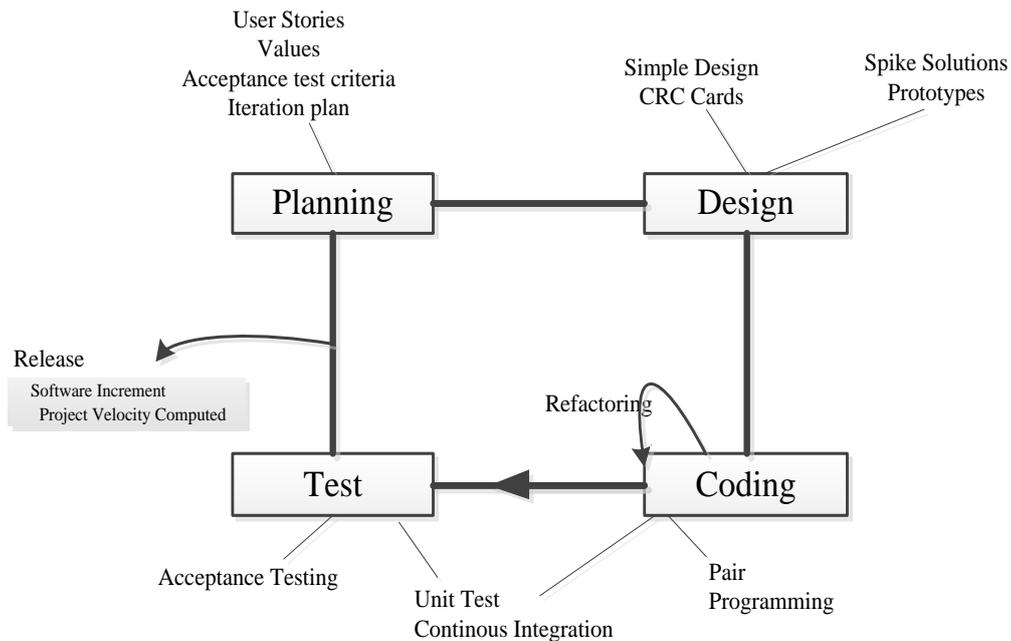
No.	Uraian	Jumlah tenaga kerja
1	Usaha Mikro	Kurang dari 5 orang
2	Usaha Kecil	5 sampai dengan 19 orang
3	Usaha Menengah	20 sampai 99 orang

2.8 Object Oriented Programming (OOP)

Object Oriented Programming (OOP) adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Berorientasi objek banyak dipilih karena metodologi lama banyak menimbulkan masalah seperti adanya kesulitan pada saat mentransformasi hasil dari satu tahap pengembangan ke tahap berikutnya. Aplikasi yang dikembangkan pada saat ini sangat beragam (aplikasi bisnis, *real-time*, *utilty*, dan sebagainya) dengan *platform* yang berbeda-beda, sehingga menimbulkan tuntutan kebutuhan metodologi pengembangan yang dapat mengakomodasi ke semua jenis aplikasi tersebut (Rosa A.S. dan Shalahuddin, 2019).

2.9 Metode *Extreme Programming*

Extreme Programming merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium serta metode ini juga sesuai jika tim dihadapkan dengan *requirement* yang tidak jelas maupun terjadi perubahan secara cepat (Suryantara, 2017).



Gambar 2.1 *Extreme Programming* (Suryantara, 2017)

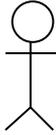
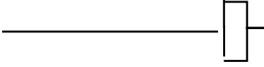
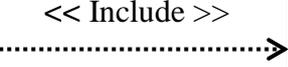
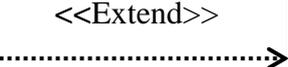
2.10 UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek, himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut (Rosa A.S. dan Shalahuddin, 2019).

2.10.1 *Use Case Diagram*

Diagram *use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat (Rosa A.S. dan Shalahuddin, 2019). Berikut simbol *use case diagram* pada Tabel 2.4.

Tabel 2.4 Simbol *Use Case Diagram* (Rosa dan Salahuddin, 2019)

No	Simbol	Deskripsi
1.		<i>Use case</i> : Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
2.		Aktor: seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda
3.		Asosiasi (<i>association</i>): merupakan komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.		Generalisasi (<i>generalization</i>): merupakan hubungan (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum
5.		Include berarti <i>use case</i> yang ditambahkan akan dipanggil saat <i>use case</i> tambahan dijalankan.
6.		Ekstensi (<i>extend</i>) merupakan <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

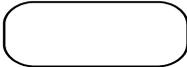
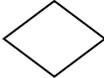
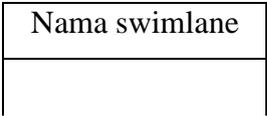
2.10.2 Activity Diagram

Activity Diagram digunakan untuk mendefinisikan atau mengelompokan aliran tampilan dari sistem tersebut (Rosa A.S. dan Shalahuddin, 2019). Berikut simbil *activity diagram* pada Tabel 2.5.

Tabel 2.5 Simbol *Activity Diagram* (Rosa dan Salahuddin, 2019)

No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.

Tabel 2.5 Simbol *Activity Diagram* (Lanjutan) (Rosa dan Salahuddin, 2019)

2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan (<i>Decision</i>) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan (<i>Join</i>) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.		Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.
6.		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

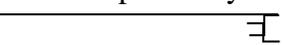
2.10.3 Class Diagram

Diagram *class* pada UML yang menggambarkan dengan jelas struktur serta deskripsi *class*, *atribut*, metode, dan hubungan dari setiap objek (Rosa A.S. dan Shalahuddin, 2019). Berikut simbol *class diagram* pada Tabel 2.6.

Tabel 2.6 Simbol *Class Diagram* (Rosa dan Salahuddin, 2019)

No.	Simbol	Deskripsi
1.		Kelas pada struktur sistem.
2.	Antar Muka/Interface  Nama_Interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	Asosiasi / Association 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan symbol

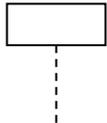
Tabel 2.6 Simbol *Class Diagram* (Lanjutan) (Rosa dan Salahuddin, 2019)

4.	Asosiasi Berarah / Directed Association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan symbol.
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	Ketergantungan / dependency 	Relasi antar kelas dengan makna ketergantungan antar kelas.
7.	Agregasi / aggregation 	Relasi antar kelas dengan maksna semua bagian (<i>whole-part</i>)

2.10.4 *Sequence Diagram*

Diagram *Sequence* menjelaskan bagaimana suatu operasi itu dilakukan; message (pesan) apa yang dikirim dan kapan pelaksanaannya (Rosa A.S. dan Shalahuddin, 2019). Berikut simbil *sequence diagram* pada Tabel 2.7.

Tabel 2.7 Simbol *Sequence Diagram* (Rosa dan Salahuddin, 2019)

No.	Simbol	Deskripsi
1.	<i>Object lifeline</i> 	Menggambarkan panjang kehidupan suatu objek selama scenario sedang di buat contohnya
2.	<i>Activation</i> 	Dimana proses sedang dilakukan oleh <i>object</i> atau <i>class</i> untuk memenuhi pesan atau perintah
3.	<i>Message</i> 	Sebuah anak panah yang mengindikasikan pesan diantara objek. Dan objek dapat mengirimkan pesan ke dirinya sendiri

2.11 *Black Box Testing*

Black Box Testing merupakan pengujian fungsional struktur program tidak dipertimbangkan. Kasus uji diputuskan berdasarkan persyaratan atau spesifikasi program atau modul dan internal yang diuji kepada user. Metode *Black box Testing* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang output pakai. Proses *Black Box Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh (Ningrum *et al.*, 2019).