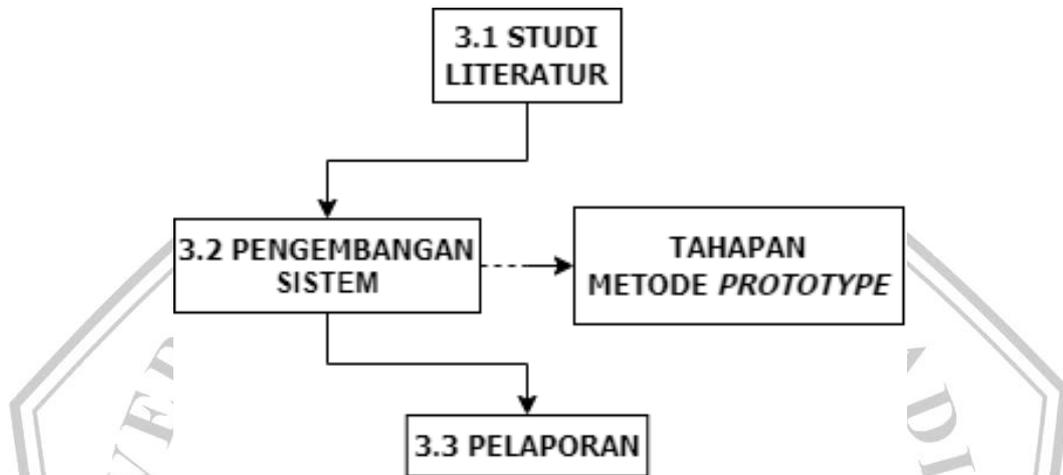


BAB III METODOLOGI PENELITIAN

Berikut ini beberapa tahapan penelitian yang akan dilakukan mencakup beberapa langkah seperti pada gambar 3.1.



Gambar 3. 1 Tahapan Penelitian

3.1 Studi Literatur

Studi literatur merupakan langkah penting dalam penelitian yang membantu seorang peneliti memahami landasan teoritis pada penelitian lain maupun sebelumnya yang relevan dengan topik penelitian saat ini. Mengumpulkan sumber-sumber yang berkaitan dengan masalah penelitian, memungkinkan untuk mengidentifikasi celah pengetahuan dalam merancang penelitian dengan lebih baik. Beberapa sumber dikumpulkan seperti jurnal ilmiah, buku, artikel, dan sumber elektronik lainnya terkait topik penelitian.

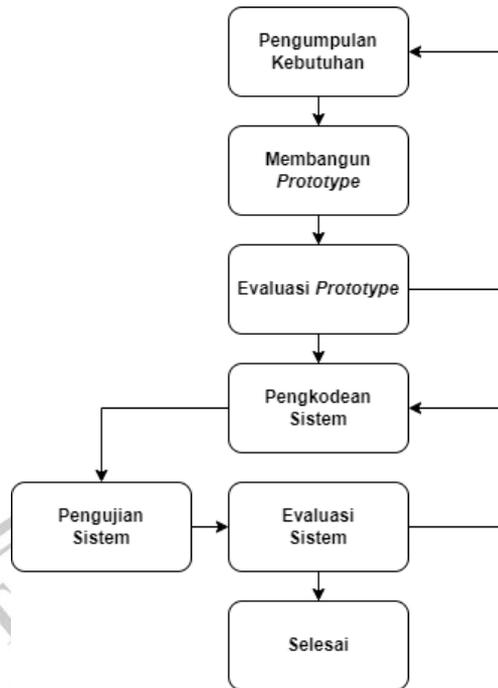
3.2 Pengembangan Sistem

Penelitian ini menggunakan metode prototype yang merupakan sebuah metode pengembangan software yang cukup banyak digunakan oleh para peneliti. Metode prototipe adalah teknik pengembangan sistem yang memungkinkan pengembang dan pelanggan berinteraksi satu sama lain selama proses pengembangan software. Metode ini memungkinkan pengembang dan pelanggan memiliki gambaran yang jelas tentang sistem yang akan dibangun oleh tim

pengembang. Ini sangat menguntungkan dan mempermudah proses pembuatan perangkat lunak. Metode prototype digunakan oleh pengembang perangkat untuk membuat prototipe dalam bentuk model aplikasi. Metode ini cocok untuk kasus di mana pengguna tidak dapat mengatakan apa yang mereka butuhkan [23]. Karena model prototipe ini memungkinkan pencipta dan pengguna untuk berkomunikasi secara terus menerus selama proses pembuatan aplikasi dan menerima umpan balik dari pengguna, model ini akan sangat membantu kedua belah pihak dalam proses pembuatan aplikasi [24].

Dalam membangun prototype, dimulai dengan pengumpulan kebutuhan. Pengembang dan pelanggan menjalin komunikasi untuk menentukan tujuan keseluruhan dari perangkat lunak, mengidentifikasi persyaratan apa pun yang diketahui, dan garis besar dari sistem yang akan dibuat. Setelah itu, dilanjutkan dengan merancang desain yang berfokus pada kebutuhan-kebutuhan dari perangkat lunak. Desain ini mengarah pada pembangunan prototype yang sesuai dengan kebutuhan pelanggan. Setelah prototype selesai dirancang, kemudian dilanjutkan lagi dengan proses evaluasi oleh pengguna untuk menyempurnakan persyaratan perangkat lunak yang akan dikembangkan. Dengan adanya metode prototype ini pengembang dan pelanggan dapat saling berinteraksi selama proses pembuatan aplikasi, sehingga pengembang dapat dengan mudah memodelkan perangkat lunak yang akan dibuat [25]. Dalam metode prototype memiliki tahapan-tahapan dalam melakukan proses pengembangan sistem yang akan diimplementasi oleh pengembang.

Beberapa tahapan yang akan dilakukan oleh peneliti dalam pengembangan sistem seperti pada gambar 3.2.



Gambar 3. 2 Tahapan Metode *Prototype* [25]

a. Pengumpulan Kebutuhan

Peneliti disini melakukan penggalian kebutuhan dengan menggunakan beberapa metode pengumpulan data antara lain:

1.) Metode Observasi

Metode observasi ini dilakukan dengan melakukan pengamatan secara langsung aktivitas yang terjadi dipergustakaan SMK Muhammadiyah 3 Singosari. Setelah melakukan pengamatan dipergustakaan tersebut ditemukan bahwa proses administrasi yang dilakukan seperti peminjaman buku, pengembalian buku, dan pendataan buku masih menggunakan pencatatan manual pada buku sebagai media pendataannya.

2.) Metode Wawancara

Dalam metode ini peneliti melakukan wawancara langsung dengan petugas perpustakaan. Dari hasil wawancara yang dilakukan, permasalahan yang didapati yaitu pihak petugas perpustakaan masih mengalami kesulitan dalam proses pencatatan dan pengolahan data antara lain seperti data buku, data peminjaman dan pengembalian buku hingga data pengunjung perpustakaan yang masih terdapat

beberapa data yang tidak tercatat dengan baik, juga ada beberapa pengunjung perpustakaan mengalami kesulitan dalam proses pencarian buku mereka masih harus datang langsung di perpustakaan untuk melakukan pengecekan ketersediaan buku yang dibutuhkan tersebut hal tersebut terjadi karena pelayanan masih bersifat manual.

Setelah proses wawancara dan *observasi*, dilanjutkan dengan elisitasi kebutuhan, peneliti melakukan identifikasi kebutuhan dengan merumuskan kebutuhan fungsional dari sistem yang akan dibuat. Setelah itu, dilakukan pembuatan *use case diagram* dan *activity diagram*.

1. *Use Case Diagram*

Use case diagram menunjukkan interaksi yang terjadi antara sistem dan lingkungannya. Metode pendekatan berbasis objek yang dikembangkan oleh Jacobson et al. pada tahun 1990an adalah metode yang digunakan untuk menemukan kebutuhan perangkat lunak. Untuk memahami interaksi yang terjadi antara pengguna dan sistem yang akan dibangun secara menyeluruh, kasus dapat membantu [22].

2. *Activity Diagram*

Activity Diagram atau diagram aktivitas adalah sebuah diagram yang dapat memodelkan proses-proses yang dilakukan dari sebuah sistem. Dalam *Unified Modeling Language*, *diagram aktivitas* dibuat untuk menjelaskan aktivitas komputer maupun alur aktivitas dalam organisasi. *Activity Diagram* berguna untuk memberikan gambaran mengenai aliran dari setiap aktivitas dalam program yang dibuat, cara masing-masing aliran dimulai, pilihan yang dapat dibuat, dan cara masing-masing aliran berakhir [26]. Proses yang dilakukan oleh sistem digambarkan dalam aktivitas diagram. Selain itu, alur tampilan sistem digambarkan dalam aktivitas diagram. Dalam diagram aktivitas, ada komponen yang dihubungkan melalui tanda panah. Aliran kerja digambarkan dengan urutan dari atas ke bawah [17].

b. Membangun *Prototype*

Secara sederhana, *prototype* adalah gambaran awal dari sebuah system yang sedang dibangun maupun dikembangkan. Desain *prototype* digunakan untuk

memberikan gambaran awal dari pemodelan sistem yang akan dibuat kepada user sehingga dapat menyesuaikan antara kebutuhan dengan sistem yang dibuat. Prototype juga berguna untuk mengubah sebuah ide menjadi sesuatu yang lebih nyata sehingga dapat dirasakan dan diuji-coba. Dengan memiliki prototype, pengembang dan desainer dapat mendapatkan umpan balik yang maksimal, sehingga mengurangi kesalahan. Dalam proses membangun *prototype*, peneliti menggunakan platform figma sebagai media desain. Figma adalah aplikasi desain berbasis cloud dan alat prototyping untuk proyek digital yang memungkinkan penggunaannya bekerja sama dalam proyek dan bekerja dalam tim kapan saja. Hal ini dikarenakan Figma memiliki beberapa kelebihan yaitu dapat dikerjakan bersama-sama karena berbasis cloud sehingga terbentuk komunikasi yang efektif diantara anggota tim UI dan UX [27]. Figma juga memungkinkan pengguna untuk membuat prototype interaktif dari desain mereka sehingga membantu tim dalam memahami bagaimana desain akan berfungsi.

c. Evaluasi *Prototype*

Rancangan aplikasi yang sebelumnya berbentuk *mockup* selanjutnya akan dievaluasi oleh pengguna. Pada tahap ini, peneliti dan pengguna memeriksa apakah prototype memenuhi keinginan pengguna. Apabila *prototype* sudah sesuai, maka dapat lanjut ke tahapan selanjutnya yaitu pengkodean sistem, sedangkan jika belum sesuai, maka prototype harus dilakukan revisi dengan mengulangi langkah 1, 2 dan 3 sampai client menyetujui [22]. Pada tahapan ini client yang melakukan evaluasi terhadap sistem ini adalah petugas perpustakaan SMK Muhammadiyah 3 Singosari.

d. Pengkodean Sistem

Tahap selanjutnya yaitu pengkodean sistem, pada proses ini dilakukan ketika *prototype* yang telah dibuat sudah sesuai dengan keinginan user dan telah mendapatkan kesepakatan dari user yang kemudian dilanjutkan dengan pengkodean sistem guna mengimplementasikan *prototype* menjadi sistem yang sesungguhnya. Dalam penelitian ini peneliti menggunakan bahasa pemrograman php dengan framework laravel. Laravel adalah framework PHP yang menekankan kesederhanaan dan fleksibilitas dalam desain. Ini adalah framework web pengembangan MVC yang dirancang untuk mengurangi biaya pengembangan dan perbaikan serta meningkatkan produktifitas pekerjaan dengan sintaks yang bersih

dan fungsional yang mengurangi waktu implementasi [28]. Laravel adalah framework berbasis PHP yang open source yang menggunakan konsep pengontrol tampilan model [29]. Secara umum, alur kerja MVC terdiri dari pengontrol berinteraksi dengan pengguna, meneruskan perintah pengguna ke model untuk menampilkan data yang diminta, dan kemudian model memberikan data kepada pengontrol untuk ditampilkan oleh pandangan. Taylor Otwell membuat Framework Laravel, sebuah kerangka kerja open source.

e. Pengujian Sistem

Pada titik ini, pengembang menguji atau mengecek secara langsung aplikasi untuk memastikan bahwa itu memenuhi persyaratan dan berfungsi dengan baik [26]. Ada beberapa metode yang bisa digunakan untuk melakukan pengujian sistem, antara lain yaitu *white-box testing* dan *black-box testing*. Pengujian menggunakan *white-box testing* digunakan ketika menguji code program namun membutuhkan waktu yang lama serta pemahaman yang luas sedangkan pengujian menggunakan *black-box testing* digunakan ketika pengembang menguji fungsionalitas dari sistem yang dibangun dan tidak membutuhkan waktu yang lama sehingga menjadi pertimbangan dari peneliti untuk menggunakan metode *black-box testing* pada penelitian ini. Ada beberapa teknik yang ada pada *black-box testing* diantaranya yaitu *Boundary Value Analysis*, *Decision Table Testing*, *Equivalence Partitioning*, *All-pair Testing*, *State Transition Testing*, *Cause-effect graph*, *Fuzzing*, *Cause-effect graph* dan *Orthogonal array testing*. Pada penelitian ini menggunakan teknik *Boundary Value Analysis* yang memiliki keunggulan dalam menangani kondisi yang kompleks pada sebuah sistem ketika terjadi kesalahan dalam batas-batas input, seperti batas atas dan bawah dari rentang nilai yang diizinkan. *Boundary value analysis* lebih memfokuskan kepada batas nilai maksimum dan batas nilai minimum yang sulit untuk ditemukan keagalanya [30]. BVA dapat dilakukan secara generik dengan menjaga agar satu variable berada pada nilai nominal (normal atau rata-rata) dan mengijinkan variable lain dimasukkan dengan nilai maksimal. [31]. Pengujian ini dilakukan dengan membandingkan antara ekspektasi awal hasil ouput atau respon dari suatu fitur dengan hasil ouput atau respon sesungguhnya dari fitur yang telah dibangun. Contoh Pengujian dengan *boundary value analysis* dapat dilihat pada tabel 3.1.

Tabel 3. 1 Contoh Pengujian *Boundary Value Analysis*

Test Case Id	Test Case	Scenario	Expected Result	Actual Result	Status
001 (Kode setiap objek pengujian pada nilai maksimum)	Fitur A (Objek yang ingin diuji)	Masukkan Skenario pada fitur A dengan nilai maksimum	Hasil yang diinginkan pada Fitur A dengan nilai maksimum	Hasil yang terjadi setelah Fitur A dijalankan sesuai scenario dengan nilai maksimum	PASS (Apabila actual result sesuai dengan expected result berdasarkan scenario objek uji dijalankan)
002 (Kode setiap objek pengujian pada nilai minimum)	Fitur A (Objek yang ingin diuji)	Masukkan Skenario pada fitur A dengan nilai minimum	Hasil yang diinginkan pada Fitur A dengan nilai minimum	Hasil yang terjadi setelah Fitur A dijalankan sesuai scenario dengan nilai minimum	PASS (Apabila actual result sesuai dengan expected result berdasarkan scenario objek uji dijalankan)

f. Evaluasi Sistem

Selanjutnya user kembali melakukan evaluasi terhadap sistem, untuk mengetahui apakah sistem telah selesai seperti yang diharapkan. Jika sudah maka bisa lanjut ke tahap selanjutnya, namun jika tidak maka perlu dilakukan pengulangan dari tahap pengkodean sistem. Pada tahap ini menggunakan metode UAT (*User Acceptance Test*) dalam mengevaluasi sistem. UAT adalah salah satu metode evaluasi sistem untuk melihat kesesuaian antara kebutuhan pengguna dan fungsionalitas yang ditawarkan oleh sistem. Melalui evaluasi UAT ini diharapkan dapat diketahui dengan pasti apakah sistem informasi sudah sesuai dengan kebutuhan pengguna. Terdapat beberapa metode yang digunakan untuk melakukan evaluasi terhadap sistem yang telah dibangun diantaranya yaitu *usability testing* dan *user acceptance test*. *Usability Testing* lebih fokus pada pengalaman pengguna dan kemudahan penggunaan sementara UAT lebih menekankan pada kesesuaian produk dengan kebutuhan pengguna sehingga menjadi pertimbangan dari peneliti untuk memilih menggunakan metode UAT. UAT sangat baik digunakan saat pengujian berfokus pada fungsionalitas dan kegunaan. Hasil dari UAT ini akan memberikan data kepada peneliti apakah pengguna menerima dengan baik fitur dan fungsionalitas yang disediakan oleh aplikasi atau masih membutuhkan revisi [11].

3.3 Pelaporan

Tahap pelaporan merupakan tahap akhir dari penelitian yang telah dilakukan untuk merekap aktivitas yang telah dilakukan selama penelitian. Pelaporan merupakan pencatatan untuk menyampaikan hal-hal yang berhubungan dengan hasil penelitian yang telah dilakukan. Segala dokumentasi serta komponen-komponen yang ikut terlibat dalam pembuatan sistem direkap menjadi satu kesatuan dalam sebuah dokumen.