# PENGEMBANGAN DATABASE UNTUK PLATFORM APLIKASI BERBASIS WEBSITE

**LAPORAN**

**Oleh**

**Kevin Farrell Febrianus Tanamatras 2602080636**

**Marcello Irwanto Suhardi 2602082143**

**Samuel Trishawn Iglesias 2602168633**

Sebuah gambar berisi teks, Font, Grafis, logo

Description automatically generated

**Program Studi Teknik Informatika**

**School of Computer Science**

**University Bina Nusantara**

**Malang 2024**

Program Studi teknik Informatika

School of Computer Science

Laporan Tugas

**PENGEMBANGAN DATABASE UNTUK PLATFORM APLIKASI BERBASIS WEBSITE**

**Kevin Farrell Febrianus Tanamatras 2602080636**

**Marcello Irwanto Suhardi 2602082143**

**Samuel Trishawn Iglesias 2602168633**

**ABSTRAK**

**Di era digital yang semakin pesat, platform aplikasi web menjadi kebutuhan vital bagi para pengembang untuk mengunggah, mengelola, dan mendistribusikan aplikasi mereka.** Platform ini, seperti App Store dan Play Store, memudahkan pengembang menjangkau pengguna global dan memberikan akses mudah bagi pengguna terhadap berbagai aplikasi.

**Penelitian ini bertujuan merancang struktur database untuk platform aplikasi web yang memenuhi kebutuhan pengembang dan pengguna.** Platform ini akan dilengkapi database untuk menyimpan, mengelola, dan mengolah data aplikasi, pengguna, transaksi, dan interaksi dalam platform.

**Metodologi SDLC (System Development Life Cycle) digunakan dalam penelitian ini, meliputi:** perencanaan database, definisi sistem, pengumpulan dan analisis kebutuhan, desain database, pemilihan DBMS, desain aplikasi, prototyping, implementasi, konversi data, dan pengujian.

**Hasil penelitian menunjukkan bahwa platform aplikasi web dengan struktur database yang dirancang dengan baik memberikan beberapa manfaat bagi pengembang dan pengguna.** Bagi pengembang, platform ini menawarkan kemudahan dalam mengunggah, mengelola, dan mendistribusikan aplikasi, serta meningkatkan peluang monetisasi. Bagi pengguna, platform ini menyediakan akses mudah untuk menemukan, mengunduh, dan menggunakan berbagai aplikasi yang bermanfaat.

**Penelitian selanjutnya dapat berfokus pada pengembangan Antarmuka Pengguna (UI) dan keamanan untuk memberikan pengalaman pengguna yang lebih nyaman dan aman.** Hal ini dapat dilakukan dengan merancang UI yang lebih intuitif dan mudah digunakan, serta memperkuat sistem keamanan platform untuk melindungi data sensitif pengguna dan pengembang. Selain itu, pengembangan fitur-fitur baru yang inovatif dan bermanfaat juga perlu dipertimbangkan untuk meningkatkan pengalaman pengguna dan mendorong pertumbuhan ekosistem aplikasi.

**DAFTAR ISI**

[PENGEMBANGAN DATABASE UNTUK PLATFORM APLIKASI BERBASIS WEBSITE 1](#_Toc170485761)

[BAB 1 5](#_Toc170485762)

[1.1. Latar Belakang 5](#_Toc170485763)

[1.2. Rumusan Masalah 5](#_Toc170485764)

[1.3. Tujuan dan Manfaat 5](#_Toc170485765)

[1.3.1. Tujuan 5](#_Toc170485766)

[1.3.2. Manfaat 6](#_Toc170485767)

[1.4. Ruang Lingkup 6](#_Toc170485768)

[1.5. Sistematika Penulisan 6](#_Toc170485769)

[BAB 2 7](#_Toc170485770)

[2.1. Website 7](#_Toc170485772)

[2.2. Database 7](#_Toc170485773)

[2.3. MySQL 7](#_Toc170485774)

[2.4. Platform 7](#_Toc170485775)

[2.5. Aplikasi 8](#_Toc170485776)

[2.6. Database System Development Life Cycle 8](#_Toc170485777)

[2.7. Phpmyadmin 9](#_Toc170485778)

[2.8. Wireframe 9](#_Toc170485779)

[2.9. Draw.io 9](#_Toc170485780)

[2.10. Flowc*hart* 9](#_Toc170485781)

[2.11. Unified Modelling Language (UML) 9](#_Toc170485782)

[2.12. Use case diagram 9](#_Toc170485783)

[2.13. Activity Diagram 10](#_Toc170485784)

[2.14. Class diagram 10](#_Toc170485785)

[2.15. Sequence diagram 10](#_Toc170485786)

[2.16. Entity Relationship Diagram (ERD) 10](#_Toc170485787)

[BAB 3 12](#_Toc170485788)

[3.1. Metodologi 12](#_Toc170485790)

[3.1.1. Database planning 12](#_Toc170485791)

[3.1.2. System definition 12](#_Toc170485792)

[3.1.3. Requirement collection and analysis 13](#_Toc170485793)

[3.1.4. Database design 14](#_Toc170485794)

[3.1.5. DBMS Selection 14](#_Toc170485795)

[3.1.6. Application design 14](#_Toc170485796)

[3.1.7. Prototyping 15](#_Toc170485797)

[3.1.8. Implementation 16](#_Toc170485798)

[3.1.9. Data conversion and loading 17](#_Toc170485799)

[3.1.10. Testing 17](#_Toc170485800)

[3.1.11. Operational maintenance 18](#_Toc170485801)

[3.2. Analisis 18](#_Toc170485802)

[3.2.1. Analisis Kebutuhan Pengguna 18](#_Toc170485803)

[3.2.2. Kebutuhan Informasi 19](#_Toc170485804)

[3.3. Perancangan Basis Data 19](#_Toc170485805)

[3.3.1. Perancangan Basis Data Konseptual 20](#_Toc170485806)

[3.3.2. Perancangan Basis Data Logical 23](#_Toc170485807)

[3.3.3. Mengidetifikasi Atribut Dari Setiap Entitas dan Relasi 25](#_Toc170485808)

[3.3.4. Perancangan Basis Data Physical 28](#_Toc170485809)

[3.3.5. Rancangan Alur Sistem 32](#_Toc170485810)

[BAB 4 33](#_Toc170485811)

[4.1. Spesifikasi System 33](#_Toc170485813)

[4.1.1. Spesifikasi Hardware 33](#_Toc170485814)

[4.1.2. Spesifikasi Software 33](#_Toc170485815)

[4.2. Prosedur Penggunaan Aplikasi serta Fitur Aplikasi 33](#_Toc170485816)

[4.2.1. Prosedur dan fitur untuk Pengembang 33](#_Toc170485817)

[4.2.2. Prosedur dan fitur untuk Pengguna 34](#_Toc170485818)

[4.3. Hasil dan Pembahasan Inti 35](#_Toc170485819)

[4.3.1. Struktur Database 35](#_Toc170485820)

[4.3.2. Contoh Kueri 37](#_Toc170485821)

[4.3.3. Diskusi 39](#_Toc170485822)

[BAB 5 40](#_Toc170485823)

[5.1. Kesimpulan 40](#_Toc170485825)

[5.2. Saran 40](#_Toc170485826)

[References 41](#_Toc170485827)

# BAB 1

**PENDAHULUAN**

## Latar Belakang

Dalam era digital yang semakin berkembang pesat, kebutuhan akan platform yang memungkinkan para pengembang aplikasi untuk mengunggah, mengelola, dan mendistribusikan aplikasi mereka semakin mendesak. Platform semacam ini, seperti App Store dan Play Store, tidak hanya memudahkan pengembang untuk menjangkau pengguna secara global, tetapi juga memberikan pengguna akses yang lebih mudah dan terorganisir terhadap berbagai aplikasi.

Platform aplikasi berbasis web yang memungkinkan pengembang untuk mengunggah dan mengelola aplikasi mereka serta pengguna untuk mengunduh aplikasi yang mereka butuhkan memiliki banyak manfaat. Platform ini dapat menyediakan berbagai fitur seperti pengelolaan akun pengembang, peninjauan dan moderasi aplikasi, serta sistem pelaporan dan analitik untuk membantu pengembang memahami kinerja aplikasi mereka.

Pentingnya database dalam platform ini tidak dapat diabaikan. Database berperan vital dalam menyimpan, mengelola, dan mengolah data yang berhubungan dengan aplikasi, pengguna, transaksi, dan interaksi lainnya dalam platform. Database yang terstruktur dengan baik dapat memastikan bahwa data dapat diakses dengan cepat dan akurat, mendukung proses pengambilan keputusan, dan menjaga integritas serta keamanan data.

Sistem database yang digunakan dalam platform ini harus mampu menangani berbagai jenis data, mulai dari informasi aplikasi, data pengguna, hingga statistik penggunaan. Database harus dirancang untuk mengelola data dengan efisien, mendukung operasi pencarian dan pengambilan data yang cepat, serta mampu mengakomodasi pertumbuhan data seiring dengan meningkatnya jumlah pengguna dan aplikasi di platform.

Dengan adanya platform aplikasi berbasis web yang didukung oleh sistem database yang andal, diharapkan dapat tercipta ekosistem yang mendukung pertumbuhan dan inovasi di bidang pengembangan aplikasi. Pengembang dapat lebih mudah mempublikasikan karya mereka, sementara pengguna dapat dengan mudah menemukan dan menggunakan aplikasi yang mereka butuhkan, yang pada akhirnya akan meningkatkan kepuasan dan pengalaman pengguna.

## Rumusan Masalah

Berdasarkan latar belakang yang sudah penulis jelaskan diatas, maka rumusan masalah yang dapat penulis ambil sebagi berikut:

1. Bagaimana merancang struktur database untuk platform aplikasi berbasis web?
2. Fitur apa saja yang harus disediakan oleh aplikasi platform aplikasi ini untuk memenuhi kebutuhan pengembang dan pengguna aplikasi?

## Tujuan dan Manfaat

Adapun tujuan serta manfaat dalam pembuatan aplikasi manajemen stok barang salon adalah sebagai berikut:

### Tujuan

1. Membuat database dari platform aplikasi yang memiliki atribut-atribut yang sesuai dan memiliki fungsi menambahkan, mengubah dan menghapus data dari database tersebut.
2. Mempelajari fitur-fitur apa saja yang diperlukan platform aplikasi berbasis web sehingga kebutuhan para pengguna dan pengembang dapat terpenuhi

### Manfaat

1. **Bagi Pengembang:**

* Meningkatkan Aksesibilitas Global
* Mempermudah Publikasi Aplikasi
* Meningkatkan Visibilitas Aplikasi
* Mendapatkan Umpan Balik Pengguna
* Menganalisis Performa Aplikasi
* Mempermudah Monetisasi Aplikasi

1. **Bagi Pengguna:**

* Akses Mudah ke Berbagai Aplikasi
* Pengalaman Pengunduhan Aman
* Penemuan Aplikasi Relevan
* Membaca Ulasan Pengguna
* Memantau Pembaruan Aplikasi
* Melakukan Pembayaran Aman

1. **Manfaat Umum:**

* Meningkatkan Inovasi
* Memperluas Akses Terhadap Teknologi
* Meningkatkan Efisiensi dan Produktivitas
* Mendukung Perekonomian Digital

## ****Ruang Lingkup****

Penelitian ini akan fokus pada pengembangan platform aplikasi berbasis web yang memungkinkan pengembang untuk mengunggah dan mengelola aplikasi mereka, serta pengguna untuk mengunduh aplikasi yang mereka butuhkan. Platform ini akan dilengkapi dengan database yang dirancang untuk menyimpan, mengelola, dan mengolah data yang berhubungan dengan aplikasi, pengguna, transaksi, dan interaksi lainnya dalam platform.

## Sistematika Penulisan

Laporan ini disusun dengan sistematika penulisan sebagai berikut:

Bab 1: Pendahuluan

Bab 2: Tinjauan Pustaka

Bab 3: Metodologi Penelitian

Bab 4: Hasil dan Pembahasan

Bab 5: Kesimpulan dan Saran

Daftar Pustaka

# BAB 2

**TINJAUAN PUSTAKA**



## Website

Menurut Abdullah (2016), Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia.

Website hadir dalam berbagai jenis, mulai dari website informasi yang edukatif, website e-commerce untuk bertransaksi, website bisnis untuk mempromosikan produk, hingga website media sosial untuk membangun komunitas. Masing-masing jenis website memiliki tujuan dan strukturnya sendiri, dengan elemen-elemen penting seperti homepage, navigasi, konten, footer, dan optimasi mobile.

## Database

Database, menurut Abdulloh (2018:103), adalah kumpulan informasi terstruktur yang tersimpan dalam komputer, layaknya gudang arsip raksasa. Database memungkinkan pengelolaan, akses, dan analisis data dengan mudah dan efisien menggunakan program komputer.

Manfaat utama database meliputi:

* Penyimpanan data terstruktur dan rapi
* Kemudahan akses dan pengambilan data
* Analisis data untuk menghasilkan informasi berharga
* Perlindungan data dengan kontrol akses dan backup

## MySQL

Menurut (Yeni Kustiyahningsih, Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL, 2011) “MySQL adalah sebuah basis data yang mengandung satu atau jumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel.Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel”.

Menurut (Astria Firman, 2016), MySQL adalah database server open-source yang cukup popular keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh praktisi untuk membangun suatu project. Adanya fasilitas API (Application Programming Interface) yang dimiliki oleh MySQL, memungkinkan bermacam–macam aplikasi komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL.

## Platform

Menurut (Anggriawan & Hasugian, 2017) platform merupakan perpaduan kerja antara perangkat keras (hardware) dan perangkat lunak (software). Dengan arti ini makan platform memberikan berbagai dapak yang baik, sebagai pelengkap seseorang dalam menjalankan sistemasiasi perangkatnya. Selain itu Platform menurut Kamus Besar Bahasa Indonesia (KBBI), platform adalah sebuah program, rencana kerja, sebuah pernyataan dari kelompok partai tentang program kebijakan, sampai dengan mimbar, pentas, atau panggung. Sedangkan menurut Wikipedia, platform adalah media atau wadah yang digunakan untuk menjalankan software.

## Aplikasi

Aplikasi, menurut Sanjaya (2015), adalah program komputer serbaguna yang dirancang untuk menyelesaikan tugas-tugas tertentu. Ibarat kotak alat digital, aplikasi hadir dalam berbagai bentuk dan fungsi, mulai dari membantu pekerjaan sehari-hari hingga memenuhi kebutuhan hiburan.

Beberapa contoh umum aplikasi:

* **Aplikasi produktivitas:** Microsoft Word, Microsoft Excel, Microsoft PowerPoint.
* **Aplikasi komunikasi:** WhatsApp, Instagram, Telegram, Gmail.
* **Aplikasi hiburan:** Spotify, YouTube, Mobile Legends.
* **Aplikasi edukasi:** Ruangguru, Kamus Besar Bahasa Indonesia, Google Play Book.
* **Aplikasi bisnis:** GoPay, Stockwatch, Google Analytics.

Aplikasi umumnya tersedia di platform mobile (Android, iOS) dan desktop (Windows, macOS). Pengguna dapat mengunduh dan menginstal aplikasi dari toko aplikasi resmi (Google Play Store, Apple App Store).

## Database System Development Life Cycle

Database System Development Life Cycle (SDLC) merupakan metodologi terstruktur untuk merancang, mengembangkan, dan memelihara sistem basis data. SDLC membantu memastikan bahwa sistem basis data memenuhi kebutuhan pengguna dan organisasi secara efektif dan efisien.

SDLC umumnya terdiri dari beberapa tahapan, antara lain:

1. **Perencanaan Database:** Menentukan tujuan dan ruang lingkup sistem basis data, mengidentifikasi pengguna dan kebutuhan mereka, serta mengumpulkan persyaratan sistem.
2. **Definisi Sistem:** Menentukan model data, entitas, dan hubungan antar entitas.
3. **Pengumpulan dan Analisis Persyaratan:** Mengumpulkan data yang dibutuhkan untuk sistem basis data dan menganalisisnya untuk menentukan struktur data yang optimal.
4. **Desain Database:** Merancang database secara logis dan fisik, termasuk skema, tabel, dan indeks.
5. **Pemilihan DBMS:** Memilih sistem manajemen basis data (DBMS) yang tepat berdasarkan kebutuhan dan persyaratan sistem.
6. **Desain Aplikasi:** Merancang aplikasi yang akan berinteraksi dengan sistem basis data, termasuk desain antarmuka pengguna dan logika bisnis.
7. **Prototyping (opsional):** Membangun prototipe sistem basis data untuk menguji desain dan mendapatkan umpan balik dari pengguna.
8. **Implementasi:** Mengubah desain sistem basis data menjadi sistem yang berfungsi, termasuk membangun tabel, memuat data, dan mengembangkan aplikasi.
9. **Konversi dan Pemuatan Data:** Mengkonversi data lama ke format baru dan memuatnya ke dalam sistem basis data baru.
10. **Pengujian:** Menguji sistem basis data dan aplikasi untuk memastikan fungsionalitas, kinerja, dan keamanan yang benar.
11. **Pemeliharaan Operasional:** Memantau dan memelihara sistem basis data, termasuk menerapkan patch, pembaruan, dan pencadangan data.

Penerapan SDLC menawarkan beberapa manfaat, di antaranya:

* Meningkatkan peluang keberhasilan proyek
* Memastikan kepuasan pengguna
* Meningkatkan kualitas sistem basis data
* Mengurangi biaya pengembangan

## Phpmyadmin

PhpMyAdmin, software gratis berbasis PHP, hadir sebagai solusi populer untuk mengelola tabel dan data pada database melalui web. Mendukung berbagai operasi database seperti MySQL dan MariaDB, PhpMyAdmin mumpuni dalam mengelola database, tabel, kolom, indeks, pengguna, izin, dan lainnya. Semua ini dapat dilakukan dengan mudah melalui antarmuka pengguna yang ramah, dan bagi pengguna yang lebih mahir, eksekusi pernyataan dan kueri MySQL secara langsung pun dimungkinkan.

## Wireframe

Wireframe merupakan sebuah sketsa kerangka yang dibuat untuk menata elemen website atau aplikasi. Wireframe dibaut sebelum tahap pengembangan dan berguna membantu merancang tata letak teks, gambar, dan elemen lainnya. Wirefram dibuat oleh UI/UX Designe rdengan menampilkan kotak dan garis untuk mengatur struktur visual pada kertas maupun dofware khusus. Manfaatnya, developer dapat lebih mudah memahami arah pengembangan dan terhindar dari revisi besar di kemudian hari, sehingga menghemat waktu dan memperlancar penyelesaian proyek.

## Draw.io

Draw.io adalah sebuah aplikasi yang menawarkan kemudahan dalam membuat diagram secara melalui website dari Draw.io. Draw.io memungkinkan pengguna untuk membuat UML (Unified Model Language) seperti use case, activity diagram, sequence diagram yang dibutuhkan pengguna. Aplikasi Draw.io dapat diakses di browser manapun, termasuk Google Crome yang juga memungkinkan Draw.io yang terintergrasi dengan Google Drive untuk memumpan diagram yang telah dibuatr.

## Flowc*hart*

Flowchart adalah alat visual yang digunakan untuk merepresentasikan alur kerja atau proses dalam bentuk diagram. Dalam dunia pemrograman dan sistem, flowchart digunakan untuk merencanakan, menganalisis, dan memahami langkah-langkah yang diperlukan dalam menyelesaikan suatu tugas atau masalah.

## Unified Modelling Language (UML)

Unified Modeling Language (UML), merupakan bahasa visual standar yang memiliki serangkaian diagram terintegrasi dan dibuat untuk membantu pengembang sistem dan perangkat lunak untuk menentukan, mengambarkan, membangun, dan mendokumentasikan kerangka dari sebuah sistem perangkat lunak, serta untuk memodelkan bisnis dan sistem non-perangkat lunak lainnya. UML mewakili kumpulan praktik rekayasa terbaik yang telah terbukti berhasil dalam pemodelan sistem yang besar dan kompleks.

## Use case diagram

Use case diagram merupakan alat dalam pembuatan desain sistem yang merepresentasikan tampilan tentang laur dari interaksi pengguna dengan sistem. Fungsinya adalah sebagai bluprint untuk memahami kebutuhan fungsional sistem dari sudut panadang pengguna, membantu stakeholder dalam berkomunikasi dan sebagai panduan dalam proses pengembangan.

## Activity Diagram

Activity diagram (AD) merupakan diagram UML yang menggambarkan proses dalam suatu sistem. AD menggambarkan aliran aktivitas, Langkah-langkah, serta tindakan yang terlibat dalam proses tertentu. AD membantu developer memahami proses dan mendefinisikan urutan aktivitas, serta menangkap interaksi data dalam sistem.

## Class diagram

Menurut Wira et al. (2019), class diagram merupakan alat bantu penting dalam merancang platform aplikasi berbasis web. Diagram ini menggambarkan struktur sistem secara visual, menunjukkan kelas-kelas yang ada dalam sistem, atribut yang dimiliki oleh setiap kelas, serta hubungan antar kelas. Dengan menggunakan class diagram, pengembang dapat lebih mudah memahami struktur sistem, mengidentifikasi potensi masalah, meningkatkan kualitas desain sistem, dan menghasilkan platform aplikasi yang memenuhi kebutuhan pengguna.

## Sequence diagram

Sequence diagram merupakan diagram UML yang menggambarkan interaksi antar objek dalam sistem melalui pertukaran pesan. Diagram ini menunjukkan urutan dan waktu pengiriman pesan antar objek sehingga memudahkan developer untuk memahami alur interaksi dalam sistem.

## Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah sebuah model visual yang digunakan untuk mendesain database. ERD menggambarkan hubungan antar objek atau entitas berserta detailnya dalam sebuah system untuk membantu dalam merancang database. ERD menggunakan notasi khusus untuk menggambarkan entitas, atribut, dan relasi yang ditampilkan dibawah sebagai berikut:

|  |  |
| --- | --- |
| Notasi | Keterangan |
| A white rectangular frame with black text  Description automatically generated | Entitas digambarkan sebagai persegi panjang dan melambangkan objek dalam ssstem |
| A black rhombus with black text  Description automatically generated | Relasi digambarkan sebagai bentuk *diamond* dan melambangkan hubungan antar entitas. |
| A black and white oval with black text  Description automatically generated | Atribut digambarkan sebagai lingkaran dan melambangkan atribut yang berada dalam sebuah entitas |
| A black line with a white background  Description automatically generated | Garis digunakan sebagai penghubung antara relasi dengan entitas atau relasi dan entitas dengan atribut. |

# BAB 3

**METODE PENELITIAN**



## Metodologi

Metodologi yang digunakan untuk merancang database PirtApps adalah SDLC (System Development Life Cycle), yang terdiri dari:

### Database planning

Pada tahap ini, Langkah pertama yang dilakukan adalah mengidentifikasi kebutuhan dasar dan tujuan dari database yang akan dibuat. Setelah tujuan didapatkan, penulis akan mengidentifikasi Langkah-langkah yang akan dilakukan untuk mencapai tujuan tersebut.

Tujuan pembuatan database platform aplikasi adalah untuk mengelola data game dan pengguna secara efisien. Langkah-langkah yang dilakukan adalah sebagai berikut:

* Mengumpulkan kebutuhan dari pengguna dan bisnis untuk menentukan fitur utama yang diperlukan, seperti pengelolaan game, pengguna, transaksi, dan ulasan.
* Membuat desain logis dan fisik dari database, termasuk diagram ERD (Entity-Relationship Diagram) dan skema fisik yang mencakup tabel, indeks, kunci utama, dan kunci asing.
* Membuat prototipe awal untuk menguji konsep dan mendapatkan *feedback* awal dari pengguna.
* Mengembangkan dan mengimplementasikan sistem berdasarkan desain yang telah dibuat.
* Melakukan pengujian menyeluruh untuk memastikan sistem berfungsi sesuai dengan spesifikasi, termasuk uji unit, integrasi, sistem, dan penerimaan pengguna.

### System definition

Tahap ini mendefinisikan ruang lingkup, batasan, dan aktor sistem. Sistem database PirtApps akan mencakup data game dan pengguna, tidak termasuk data keuangan atau pribadi sensitif. Aktor sistemnya adalah pengembang game, administrator platform, dan pengguna platform.

### Requirement collection and analysis

Kebutuhan pengguna dikumpulkan melalui survei, dan observasi terhadap platform aplikasi sejenis seperti Steam, Appstore dan Playstore. Kebutuhan dianalisis untuk menentukan kebutuhan informasi yang harus dipenuhi oleh database. Persyaratan sistem kemudian ditetapkan berdasarkan kebutuhan tersebut. Hasil dari analisa ditampilkan sebagai berikut.

Sebuah gambar berisi teks, cuplikan layar, Font, dokumen

Description automatically generated

Fig. 1 User requirement dan analisis

**Hasil Analisis Kebutuhan:**

##### Kebutuhan Fungsional:

###### Sistem harus menyediakan fitur untuk mengelola game, termasuk mengunggah game, memperbarui game, menghapus game, dan mengatur informasi game.

###### Sistem harus menyediakan fitur untuk mengelola pengguna, termasuk mendaftarkan pengguna, login pengguna, mengelola profil pengguna, dan mengatur hak akses pengguna.

###### Sistem harus menyediakan fitur untuk mengelola transaksi, termasuk pencatatan pembelian game, pencatatan unduhan game, dan pencatatan ulasan game.

###### Sistem harus menyediakan fitur untuk mencari game berdasarkan kategori, nama, dan harga.

###### Sistem harus menyediakan fitur untuk melihat detail game, termasuk screenshot game, deskripsi game, dan ulasan game.

###### Sistem harus menyediakan fitur untuk memberikan ulasan dan rating game.

##### Kebutuhan Non-Fungsional:

###### Sistem harus aman dan terlindungi dari akses yang tidak sah.

###### Sistem harus memiliki performa yang tinggi dan dapat diakses dengan cepat.

###### Sistem harus mudah digunakan dan dipahami oleh pengguna.

###### Sistem harus dapat dipelihara dengan mudah dan dapat diubah sesuai dengan kebutuhan.

### Database design

Perancangan basis data dilakukan dalam 3 tahap yaitu perancangan basis data konseptual, perancangan basis data logical dan perancangan basis data fisikal.

### DBMS Selection

Pada tahap ini, penulis akan memilih DBMS untuk membantu pembuatan aplikasi basis data. DBMS yang digunakan adalah MySQL

### Application design

Design Low-Fidelity dilakukan dengan menggunakan canva untuk membantu mendapatkan gambaran tentang prototype yang akan dibuat.

Sebuah gambar berisi teks, cuplikan layar, Font, garis

Description automatically generated

Fig. 2 Desain Tampilan halaman login

Sebuah gambar berisi teks, cuplikan layar, nomor, struk

Description automatically generated

Fig. 3 Desain Tampilan halaman Developer

A screenshot of a store

Description automatically generated

Fig. 4 Desain Tampilan halaman user

Sebuah gambar berisi teks, cuplikan layar, Font, diagram

Description automatically generated

Fig. 5 Desain Tampilan halaman user

### Prototyping

Prototipe aplikasi dibuat untuk pengujian dan penyempurnaan desain aplikasi.

Sebuah gambar berisi teks, cuplikan layar, Font, garis

Description automatically generated

Fig. 6 Tampilan halaman login

A screenshot of a computer

Description automatically generatedA white background with black text

Description automatically generated

Fig. 7 Tampilan halaman Developer

A screenshot of a computer

Description automatically generated

Fig. 8 Tampilan halaman User

### Implementation

Pada tahap ini, dilakukan implementasi basis data berdasarkan *database planning* yang sudah dibuat. Implementasi database PirtApps meliputi:

**1. Membuat struktur tabel database:** Struktur tabel database dibuat sesuai dengan desain logis yang telah dibuat pada tahap 3.3.4 Perintah SQL CREATE TABLE digunakan untuk membuat tabel-tabel dalam database.

**2. Membuat hubungan antar tabel:** Hubungan antar tabel dibuat dengan menggunakan kunci utama (primary key) dan kunci asing (foreign key). Kunci utama digunakan untuk mengidentifikasi baris unik dalam tabel, sedangkan kunci asing digunakan untuk menghubungkan tabel yang berbeda.

**3. Memasukkan data ke dalam database:** Data game dan pengguna dimasukkan ke dalam database menggunakan perintah SQL INSERT INTO. Data dapat dimasukkan secara manual atau melalui proses import dari file eksternal.

**3. Membuat prosedur dan fungsi:** Prosedur dan fungsi dibuat untuk mengotomatisasi tugas-tugas yang berulang, seperti menghitung total pendapatan dari penjualan game atau memberikan rekomendasi game kepada pengguna.

### Data conversion and loading

Konversi dan pemuatan data dilakukan untuk memindahkan data game dan pengguna dari format lama ke format baru yang sesuai dengan struktur database PirtApps. Hal ini dapat dilakukan dengan menggunakan berbagai metode, seperti:

**Import data dari file CSV:** Data game dan pengguna dapat diimpor dari file CSV ke dalam database menggunakan perintah SQL LOAD DATA INFILE.

**Konversi data manual:** Data game dan pengguna dapat dikonversi secara manual ke format baru dan kemudian dimasukkan ke dalam database menggunakan perintah SQL INSERT INTO.

### Testing

Pada tahap ini, dilakukan pengujian untuk memastikan bahwa database PirtApps berfungsi dengan baik dan sesuai dengan spesifikasi yang telah ditentukan. Pengujian yang dilakukan dalam penelitian ini adalah **pengujian manual** dengan cara menginput data game ke dalam database dan mengamati hasilnya.

**Langkah-langkah pengujian:**

**1. Siapkan data uji game:** Siapkan data uji game yang sesuai dengan skenario pengujian yang ingin dilakukan. Data uji game dapat berupa nama game, deskripsi game, kategori game, harga game, dan tanggal rilis.

**2. Masukkan data uji game ke dalam database:** Masukkan data uji game ke dalam database menggunakan perintah SQL INSERT INTO.

**3. Lakukan operasi database terkait game:** Lakukan operasi database terkait game sesuai dengan skenario pengujian, seperti:

###### **Menambahkan game baru:** Jalankan query SQL INSERT INTO untuk menambahkan game baru ke dalam database.

###### **Memperbarui informasi game:** Jalankan query SQL UPDATE untuk memperbarui informasi game yang sudah ada di database.

###### **Menghapus game:** Jalankan query SQL DELETE untuk menghapus game dari database.

###### **Mencari game:** Jalankan query SQL SELECT untuk mencari game berdasarkan nama, kategori, harga, atau tanggal rilis.

**4. Amati hasil operasi:** Amati hasil operasi database dan bandingkan dengan hasil yang diharapkan. Jika hasil yang diamati tidak sesuai dengan hasil yang diharapkan, catat error yang terjadi.

**5. Ulangi langkah 1-4:** Ulangi langkah 1-4 untuk skenario pengujian yang berbeda.

### Operational maintenance

Pemeliharaan operasional adalah aktivitas yang dilakukan secara berkelanjutan untuk memastikan database PirtApps tetap berfungsi dengan baik dan memenuhi kebutuhan pengguna. Kegiatan pemeliharaan operasional meliputi:

**1. Pencadangan data (data backup):** Pencadangan data dilakukan secara rutin untuk melindungi data dari kehilangan akibat kerusakan hardware, software, atau bencana alam. Cadangan data dapat disimpan di lokasi yang berbeda untuk keamanan lebih.

**2. Pemulihan data (data recovery):** Jika terjadi kerusakan data, pemulihan data dapat dilakukan menggunakan cadangan data yang terakhir dibuat.

## Analisis

### Analisis Kebutuhan Pengguna

Platform aplikasi berbasis web ini harus memenuhi kebutuhan berbagai jenis pengguna, termasuk pengembang dan pengguna aplikasi. Berikut adalah beberapa kebutuhan pengguna yang perlu dipertimbangkan:

**Manajemen Aplikasi:**

##### Pengembang:

###### Mengunggah, mengelola, dan memperbarui aplikasi mereka.

###### Memantau performa aplikasi dan mendapatkan laporan penggunaan.

###### Mengelola akun developer dan mengatur pengaturan profil.

###### Berkomunikasi dengan tim platform dan pengguna aplikasi.

##### Pengguna:

###### Menjelajahi dan menemukan aplikasi baru.

###### Membaca deskripsi aplikasi, melihat tangkapan layar, dan menonton video demo.

###### Membaca ulasan dan penilaian pengguna lain.

###### Mengunduh dan menginstal aplikasi.

###### Memperbarui aplikasi yang sudah terpasang.

###### Mengelola langganan aplikasi dan item dalam aplikasi.

**Pencarian dan Navigasi:**

##### Pengguna:

###### Mencari aplikasi berdasarkan judul, kata kunci, kategori, atau tag.

###### Menelusuri aplikasi berdasarkan genre, popularitas, harga, rating, dan tanggal rilis.

###### Memfilter hasil pencarian berdasarkan berbagai kriteria.

###### Menemukan aplikasi yang direkomendasikan berdasarkan minat dan riwayat penggunaan mereka.

**Pembelian dan Pembayaran:**

##### Pengguna:

###### Membeli aplikasi, langganan, atau item dalam aplikasi dengan aman dan mudah.

###### Menggunakan berbagai metode pembayaran yang didukung, seperti kartu kredit, dompet digital, dan transfer bank.

###### Melacak riwayat pembelian dan mengelola langganan mereka.

###### Mendapatkan pengembalian dana atau penukaran untuk aplikasi yang tidak sesuai.

**Ulasan dan Penilaian:**

##### Pengguna:

###### Memberikan ulasan dan penilaian untuk aplikasi yang telah mereka gunakan.

###### Membaca ulasan dan penilaian dari pengguna lain untuk membantu mereka dalam pengambilan keputusan.

###### Melaporkan ulasan dan penilaian yang tidak pantas atau menipu.

### Kebutuhan Informasi

Platform aplikasi berbasis web ini perlu menyimpan dan mengelola berbagai jenis informasi, termasuk:

###### Informasi aplikasi: Judul, deskripsi, kata kunci, kategori, tag, tangkapan layar, video demo, harga, rating, ulasan, dan informasi pengembang.

###### Informasi pengguna: Nama, alamat email, kata sandi, informasi pembayaran, riwayat pembelian, dan langganan.

###### Informasi transaksi: Detail pembelian, seperti tanggal, waktu, jumlah, dan metode pembayaran.

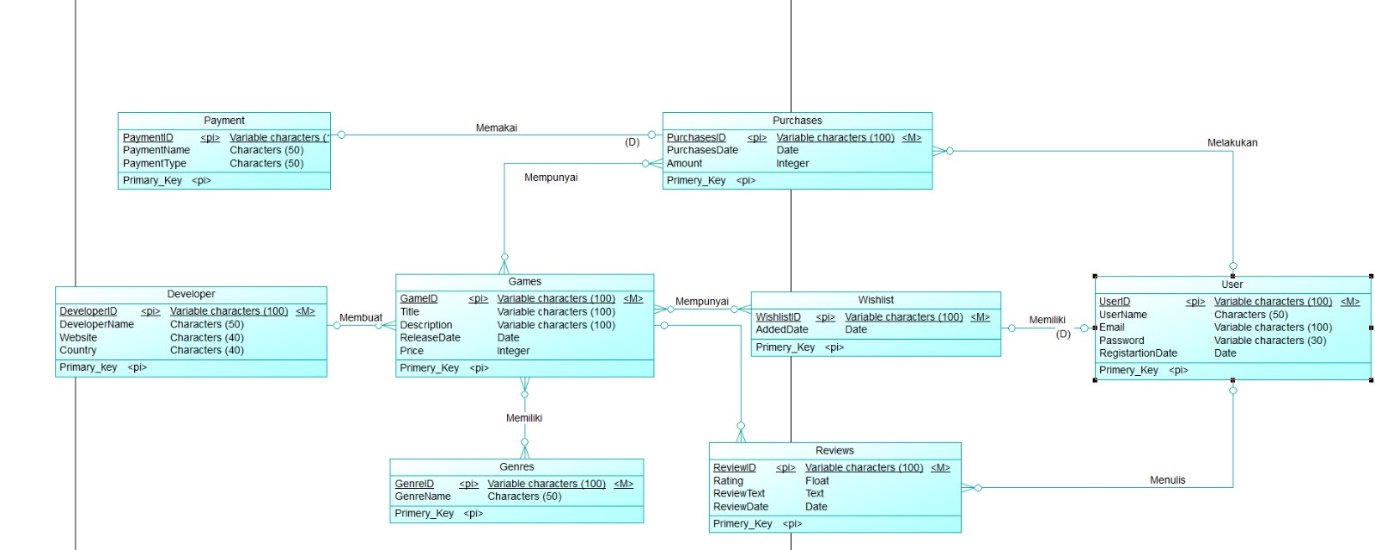
###### Data penggunaan aplikasi: Data tentang bagaimana pengguna berinteraksi dengan aplikasi, seperti waktu penggunaan, fitur yang digunakan, dan data crash.

## Perancangan Basis Data

Dalam melakukan perancangan basis data, terdapat tiga tahapan penting yang harus dilakukan, yaitu:

* 1. Perancangan basis data konseptual.
  2. Perancangan basis data logikal.
  3. Perancangan basis data fisikal.

### Perancangan Basis Data Konseptual



A diagram of a company

Description automatically generated

Fig. 9 ERD

#### Entitas dan Atribut

|  |  |
| --- | --- |
| Users:  UserID (Primary Key) Varchar [100]  Username Char [50]  Email Varchar [100]  Password Varchar [30]  Registration Date Date [40]  Last Login Date Date [40]  Games:  GameID (Primary Key) Varchar [100]  Title Varchar [50]  Description Varchar [100]  Release Date Date [40]  DeveloperID (Foreign Key) Varchar [100]  Price Integer [100]  Genre Char [50]  Developers:  DeveloperID (Primary Key) Varchar [100]  DeveloperName Char [50]  Website Char [40]  Country Char [10]  Payment:  PaymentID (Primary Key) Varchar [100]  PurchaseID (Foreign Key) Varchar [100]  Name\_Payment Char [10]  Type\_of\_Payment Char [10] | Genres:  GenreID (Primary Key) Varchar [100]  GenreName Char [50]  Purchases:  PurchaseID (Primary Key) Varchar [100]  UserID (Foreign Key) Varchar [100]  GameID (Foreign Key) Varchar [100]  Purchase Date Date [40]  Amount Integer [100]  Reviews:  ReviewID (Primary Key) Varchar [100]  UserID (Foreign Key) Varchar [100]  GameID (Foreign Key) Varchar [100]  Rating Float [10]  Review Text Char [100]  Review Date Date [50]  Wishlist:  WishlistID (Primary Key) Varchar [100]  UserID (Foreign Key) Varchar [100]  GameID (Foreign Key) Varchar [100]  AddedDate Date [50] |

#### Mengidentifikasi tipe relasi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nama Entitas | Multiplicity | Relationship | Multiplicity | Nama Entitas |
| User | 1..\* | Can do | 1..1 | Purchases |
| User | 1..1 | Have | 1..1 | Wishlist |
| User | 1..\* | Write | 1..1 | Reviews |
| Purchases | 1..1 | Use | 1..1 | Payment |
| Purchases | 1..\* | Have | 1..\* | Games |
| Developer | 1..\* | Manage | 1..1 | Games |
| Games | 1..\* | Have | 1..\* | Genre |
| Games | 1..\* | Have | 1..1 | Review |

#### Mengidentifikasi dan Mengasosiasikan Atribut Suatu Entitas atau Tipe Relasi

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Nama Entitas | Deskripsi | Kegunaan |
| 1. | Tabel User | Menyimpan data para user | Pengguna yang baru mendaftar datanya akan tersimpan pada tabel ini sehingga mereka bisa melakukan login. |
| 2. | Tabel Developer | Menympan data para developer game | Developer yang hendak menerbitkan game akan disimpan datanya pada tabel ini. |
| 3. | Tabel Games | Menyimpan data game yang dijual di paltform aplkasi | Game yang telah diterbitkan oleh developer akan disimpan pada tabel ini dan dapat dibeli oleh user. |
| 4. | Tabel Purchases | Menyimpan data transaksi pembelian game oleh user | User yang ingin membeli game dapat melakukan transaksi dan bukti transaksi akan terkumpul pada tabel ini. |
| 5. | Tabel Payment | Menympan data metode pembayaran | Pada saat transaksi berlangsung, user harus memilih payment gateaway yang telah terdaftar pada tabel payment. |
| 6. | Tabel Genre | Menympan data dari genre game | Genre dari game yang terdaftar dapat dilihat dari tabel ini |
| 7. | Tabel Wishlist | Menyimpan data game yang dinginkan oleh user | User yang menginginkan suatu game namun belum memutuskan untuk membelinya dapat menympan game tersebut di tabel wishlist mereka agar mereka tidak perlu repot mencari dan mendapatkan notifikasi pada saat gamenya sedang obral. |
| 8. | Tabel Reviews | Menyimpan data game yang direview oleh user | User yang telah memainkan suatu game daoat menuliskan review mereka dan disimpan dalam tabel reviews. |

### Perancangan Basis Data Logical

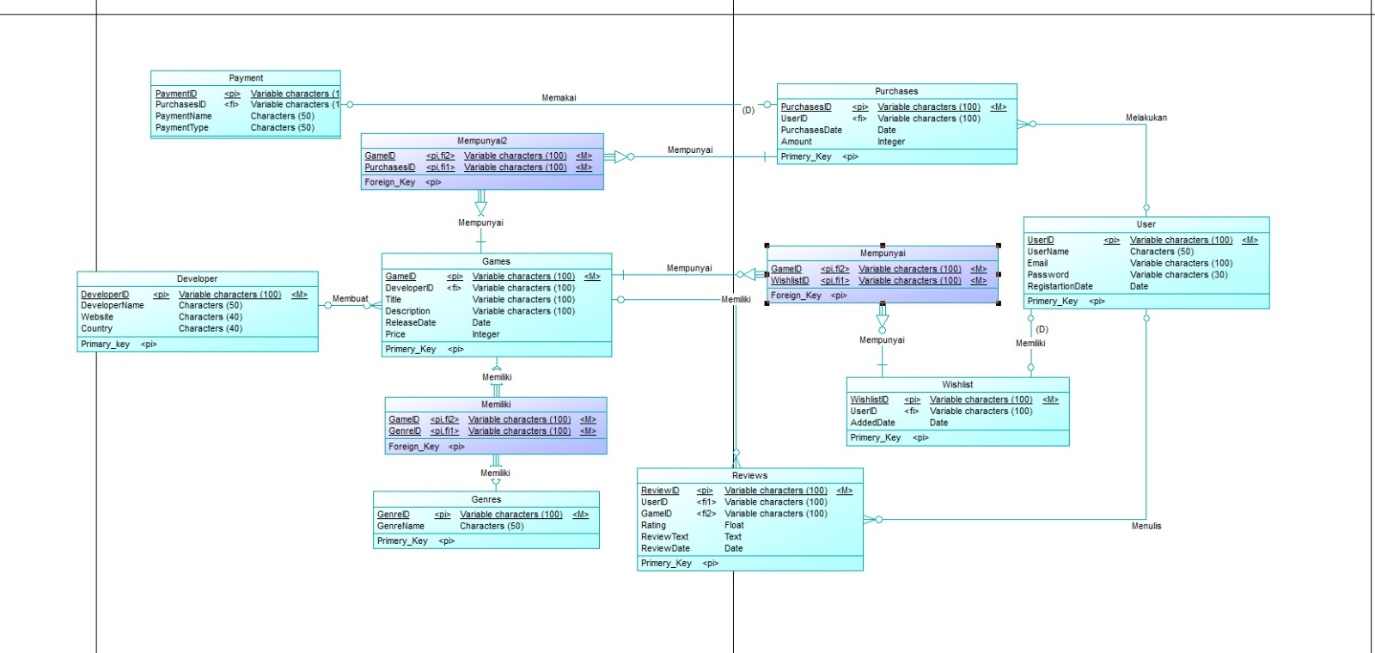


Fig. 10 Class Diagram

Berikut adalah gambaran dari setiap tabel yang ada pada Diagram Data Logical

#### Tabel User

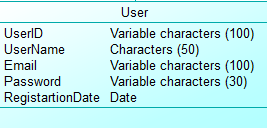


Fig. 11 Tabel User

#### Tabel Developer

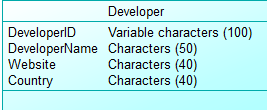


Fig. 12 Tabel Developer

#### Tabel Games

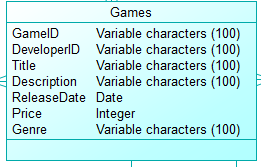


Fig. 13 Tabel Games

#### Tabel Purchases

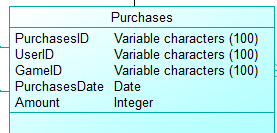


Fig. 14 Tabel Purchases

#### Tabel Payment

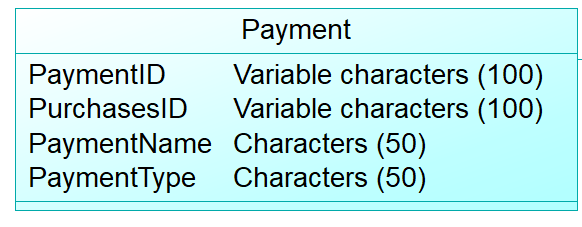


Fig. 15 Tabel Payment

#### Tabel Genre

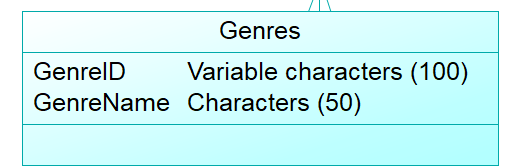


Fig. 16 Tabel Genre

#### Tabel Reviews

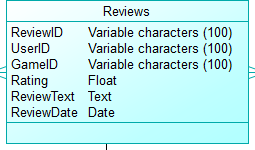


Fig. 17 Tabel Reviews

#### Tabel Wishlist

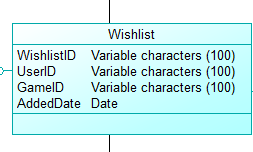


Fig. 18 Tabel Wishlist

### Mengidetifikasi Atribut Dari Setiap Entitas dan Relasi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data type &length | NULL | Multivalued |
| Tabel User | UserID (PK) | ID dari user | Variable characters (100) | NO | NO |
| UserName | Nama User | Characters (50) | NO | NO |
| Email | Email user | Variable characters (100) | NO | NO |
| Password | Password user | Variable characters (30) | NO | NO |
| RegistrationDate | Tanggal registrasi | Date | NO | NO |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data type &length | NULL | Multivalued |
| Tabel Developer | DeveloperID (PK) | ID dari game | Variable characters (100) | NO | NO |
| DeveloperName | Nama Developer | Characters (50) | NO | NO |
| Website | Alamat Website asli developer | Characters (50) | NO | NO |
| Country | Negara asal Developer | Characters (50) | NO | NO |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data type &length | NULL | Multivalued |
| Tabel Games | GamesID (PK) | ID dari game | Variable characters (100) | NO | NO |
| DeveloperID (FK) | ID dari Developer | Variable characters (100) | NO | NO |
| Title | Judul game | Variable characters (100) | NO | NO |
| Description | Penjelasan singkat dari game | Variable characters (100) | NO | NO |
| Release Date | Tanggal rilis game | Date | NO | NO |
| Price | Harga dari game | Interger | NO | NO |
| Genre | Genre dari game | Variable characters (100) | NO | NO |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | | Description | | Data type &length | | NULL | | Multivalued |
| Tabel Purchases | | PuchasesID (PK) | | ID pembelian | | Variable characters (100) | | NO | NO |
| UserID (FK) | | ID dari user | | Variable characters (100) | | NO | NO |
| GameID (FK) | | ID dari game | | Variable characters (100) | | NO | NO |
| PurchasesDate | | Tanggal pembelian game | | Date | | NO | NO |
| Amount | | Jumlah pembayaran | | Integer | | NO | NO |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data type &length | NULL | Multivalued |
| Tabel  Wishlist | WishlistID (PK) | ID Wishlist yang dibuat | Variable characters (100) | NO | NO |
|  | UserID (FK) | ID dari User | Variable characters (100) | NO | NO |
|  | GameID (FK) | ID dari Game | Variable characters (100) | NO | NO |
|  | AddedDate | Tanggal kapan Wishlist dibuat | Date | NO | NO |

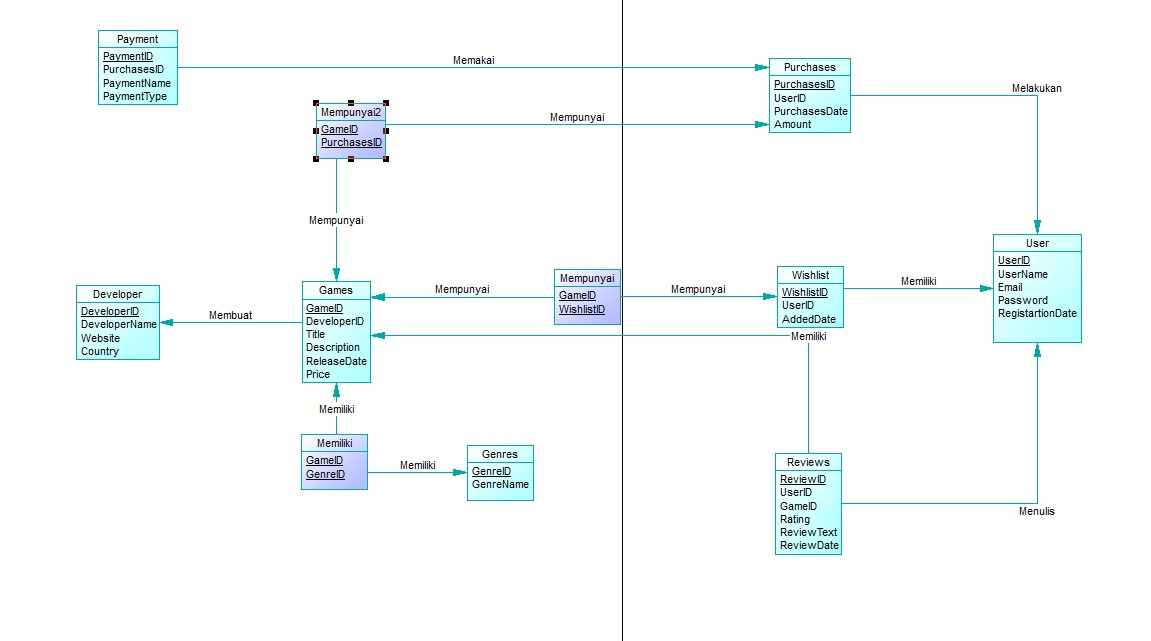
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data type &length | NULL | Multivalued |
| Tabel Reviews | ReviewID (PK) | ID review | Variable characters (100) | NO | NO |
| UserID (FK) | ID dari user | Variable characters (100) | NO | NO |
| GameID (FK) | ID dari game | Variable characters (100) | NO | NO |
| Rating | Nilai dari game yang di review | Float | NO | NO |
| ReviewText | Isi review | Text | NO | NO |
| ReviewDate | Tanggal review | Date | NO | NO |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data type &length | NULL | Multivalued |
| Tabel Payment | PaymentID (PK) | ID dari payment yang digunakan | Variable characters (100) | NO | NO |
| PuchasesID (FK) | ID dari pembelian | Variable characters (100) | NO | NO |
| PaymentName | Nama dari Payment yang dipilih | Characters (50) | NO | NO |
| PaymentType | Tipe payment yang digunakan | Characters (50) | NO | NO |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data type &length | NULL | Multivalued |
| Tabel Genres | GenreID (PK) | ID dari Genre game | Variable characters (100) | NO | NO |
|  | GenreName | Nama Genre | Characters (50) | NO | NO |

### Perancangan Basis Data Physical

Pada tahap ini, model data logis yang telah dibuat pada tahap sebelumnya (lihat Bab 3) diterjemahkan ke dalam struktur database fisik menggunakan bahasa pemrograman SQL.



#### Menerjemahkan Model Data Logikal Untuk DBMS yang Digunakan

* 1. Merancang Entitas dan Atribut
     1. Membuat Tabel Developer

CREATE TABLE `developers` (

`DeveloperID` varchar(100) NOT NULL,

`DeveloperName` char(50) NOT NULL,

`Website` char(40) NOT NULL,

`Country` char(40) NOT NULL

)

* + 1. Membuat Tabel Games

CREATE TABLE `games` (

`GameID` varchar(100) NOT NULL,

`DeveloperID` varchar(100) NOT NULL,

`Tittle` varchar(100) NOT NULL,

`Description` varchar(100) NOT NULL,

`ReleaseDate` date NOT NULL,

`Price` int(100) NOT NULL,

`Genre` varchar(100) NOT NULL

)

* + 1. Membuat Tabel Genres

CREATE TABLE `genres` (

`GenreID` varchar(100) NOT NULL,

`GenreName` char(50) NOT NULL

)

* + 1. Membuat Tabel Payment

CREATE TABLE `payment` (

`PaymentID` varchar(100) NOT NULL,

`PurchaseID` varchar(100) NOT NULL,

`PaymentName` char(50) NOT NULL,

`PaymentType` char(50) NOT NULL

)

* + 1. Membuat Tabel Purchases

CREATE TABLE `purchases` (

`PurchaseID` varchar(100) NOT NULL,

`UserID` varchar(100) NOT NULL,

`GameID` varchar(100) NOT NULL,

`PurchaseDate` date NOT NULL,

`Amount` int(100) NOT NULL

)

* + 1. Membuat Tabel Review

CREATE TABLE `reviews` (

`ReviewID` varchar(100) NOT NULL,

`UserID` varchar(100) NOT NULL,

`GameID` varchar(100) NOT NULL,

`Rating` float NOT NULL,

`ReviewText` mediumtext NOT NULL,

`ReviewDate` date NOT NULL

)

* + 1. Membuat Tabel User

CREATE TABLE `user` (

`UserID` varchar(100) NOT NULL,

`Username` char(50) NOT NULL,

`Email` varchar(100) NOT NULL,

`Password` varchar(30) NOT NULL,

`RegistrationDate` date NOT NULL,

`LastLoginDate` date NOT NULL

)

* + 1. Membuat Tabel Wishlist

CREATE TABLE `wishlist` (

`WishlistID` varchar(100) NOT NULL,

`UserID` varchar(100) NOT NULL,

`GameID` varchar(100) NOT NULL,

`AddedDate` date NOT NULL

)

* 1. Membuat Primary Key dan Foreign Key
     1. Tabel developer

ALTER TABLE `developers`

ADD PRIMARY KEY (`DeveloperID`);

* + 1. Tabel games

ALTER TABLE `games`

ADD PRIMARY KEY (`GameID`),

ADD KEY `DeveloperID` (`DeveloperID`),

ADD KEY `Genre` (`Genre`);

* + 1. Tabel genre

ALTER TABLE `genres`

ADD PRIMARY KEY (`GenreID`);

* + 1. Tabel payment

ALTER TABLE `payment`

ADD PRIMARY KEY (`PaymentID`),

ADD KEY `PurchaseID` (`PurchaseID`);

* + 1. Tabel purchases

ALTER TABLE `purchases`

ADD PRIMARY KEY (`PurchaseID`),

ADD KEY `UserID` (`UserID`),

ADD KEY `GameID` (`GameID`);

* + 1. Tabel reviews

ALTER TABLE `reviews`

ADD PRIMARY KEY (`ReviewID`),

ADD KEY `UserID` (`UserID`),

ADD KEY `GameID` (`GameID`);

* + 1. Tabel user

ALTER TABLE `user`

ADD PRIMARY KEY (`UserID`);

* + 1. Tabel wishlist

ALTER TABLE `wishlist`

ADD PRIMARY KEY (`WishlistID`),

ADD KEY `UserID` (`UserID`),

ADD KEY `GameID` (`GameID`);

* 1. Membuat auto increment yang dapat mengubungkan data antar tabel pada sistem
     1. Tabel games

ALTER TABLE `games`

ADD CONSTRAINT `DeveloperID.key` FOREIGN KEY (`DeveloperID`) REFERENCES `developers` (`DeveloperID`),

ADD CONSTRAINT `GenreGame` FOREIGN KEY (`Genre`) REFERENCES `genres` (`GenreID`);

* + 1. Tabel payment

ALTER TABLE `payment`

ADD CONSTRAINT `PurchaseID.key` FOREIGN KEY (`PurchaseID`) REFERENCES `purchases` (`PurchaseID`);

* + 1. Tabel purchases

ALTER TABLE `purchases`

ADD CONSTRAINT `GameID` FOREIGN KEY (`GameID`) REFERENCES `games` (`GameID`),

ADD CONSTRAINT `UserID.kfc` FOREIGN KEY (`UserID`) REFERENCES `user` (`UserID`);

* + 1. Tabel reviews

ALTER TABLE `reviews`

ADD CONSTRAINT `GameID.key` FOREIGN KEY (`GameID`) REFERENCES `games` (`GameID`),

ADD CONSTRAINT `UserID` FOREIGN KEY (`UserID`) REFERENCES `user` (`UserID`);

* + 1. Tabel wishlist

ALTER TABLE `wishlist`

ADD CONSTRAINT `UserID.wishlist` FOREIGN KEY (`UserID`) REFERENCES `user` (`UserID`),

ADD CONSTRAINT `Wislist\_blablabla` FOREIGN KEY (`GameID`) REFERENCES `games` (`GameID`);

COMMIT;

### Rancangan Alur Sistem

#### Flowchart

A diagram of a flowchart

Description automatically generated

Fig. 19 Flowchart

# BAB 4

**HASIL DAN PEMBAHASAN**



## Spesifikasi System

Mengembangkan sistem yang baik memerlukan spesifikasi perangkat keras (hardware) dan perangkat lunak (software) yang mumpuni juga, oleh karena itu berikut spesifikasi hardware dan software yang dibutuhkan untuk menjalankan platform aplikasi PirtApps.

### Spesifikasi Hardware

Spesifikasi hardware akan tergantung pada skala dan kompleksitas PirtApps. Namun, secara umum kira-kira spesifikasi hardware yang dibutuhkan sebagai server utama database PirtApps meliputi:

Prosesor: Minimal Intel Core i5 gen 8 atau AMD Ryzen 5 1600x

Memori: Minimal 8 GB RAM

Storage: Minimal 250 GB SSD

Jaringan: Koneksi internet stabil dengan bandwidth 10 mbps yang cukup untuk menangani lalu lintas data.

### Spesifikasi Software

Spesifikasi software yang mumpuni sangat dibutuhkan untuk menjalankan database dari PirtApps, Tetapi spesifikasi tersebut juga akan tergantung pada teknologi yang dipilih untuk pengembangan platform aplikasi. Namun, umumnya spesifikasi software yang dibutuhkan meliputi:

##### Sistem Operasi Server:

###### Pilihan umum: Linux (misalnya Ubuntu, CentOS) atau Windows Server

##### Sistem Manajemen Basis Data (DBMS):

###### Pilihan umum: MySQL, PostgreSQL

##### Bahasa Pemrograman:

###### Pilihan umum: PHP, Python, Java

##### Web Server:

###### Pilihan umum: Apache, Nginx

##### **Framework**:

###### Pilihan umum : Laravel, Boostrap

## Prosedur Penggunaan Aplikasi serta Fitur Aplikasi

Prosedur penggunaan dan fitur aplikasi sangat dibutuhkan untuk memenuhi kebutuhan pengguna dan developer. Disini prosedur dan fitur akan tergantung pada jenis pengguna (developer atau pengguna), berikut prosedur dan fitur aplikasi yang PirtApps sediakan meliputi:

### Prosedur dan fitur untuk Pengembang

##### Pendaftaran Akun Developer:

###### Akses halaman pendaftaran developer pada platform aplikasi.

###### Isi formulir pendaftaran dengan informasi yang diperlukan, seperti nama developer, alamat email, dan password.

###### Verifikasi akun melalui email atau link verifikasi yang dikirimkan.

##### Pengunggahan Aplikasi:

###### Login ke akun developer.

###### Akses halaman pengunggahan aplikasi.

###### Isi formulir pengunggahan dengan informasi aplikasi, seperti judul, deskripsi, kategori, tangkapan layar, dan file aplikasi.

###### Submit formulir pengunggahan.

##### Pengelolaan Aplikasi:

###### Login ke akun developer.

###### Akses halaman pengelolaan aplikasi.

###### Lihat daftar aplikasi yang telah diunggah.

###### Edit informasi aplikasi, seperti deskripsi, harga, dan kategori.

###### Upload pembaruan untuk aplikasi yang sudah ada.

###### Lihat laporan performa aplikasi, seperti jumlah unduhan, ulasan, dan rating.

##### Komunikasi dengan Tim Platform:

###### Login ke akun developer.

###### Akses halaman kontak tim platform.

###### Kirimkan pesan atau laporan kepada tim platform.

### Prosedur dan fitur untuk Pengguna

##### Pendaftaran Akun Pengguna:

###### Akses halaman pendaftaran pengguna pada platform aplikasi.

###### Isi formulir pendaftaran dengan informasi yang diperlukan, seperti nama pengguna, alamat email, dan password.

###### Verifikasi akun melalui email atau link verifikasi yang dikirimkan.

##### Penjelajahan Aplikasi:

###### Login ke akun pengguna.

###### Akses halaman penjelajahan aplikasi.

###### Gunakan fitur pencarian dan filter untuk menemukan aplikasi yang diinginkan.

###### Lihat daftar aplikasi yang tersedia, beserta informasi seperti judul, deskripsi, kategori, rating, dan harga.

##### Unduh dan Instalasi Aplikasi:

###### Login ke akun pengguna.

###### Akses halaman detail aplikasi yang diinginkan.

###### Klik tombol unduh untuk aplikasi gratis atau tombol beli untuk aplikasi berbayar.

###### Ikuti langkah-langkah instalasi aplikasi sesuai dengan perangkat yang digunakan.

##### Pemberian Ulasan dan Penilaian:

###### Login ke akun pengguna.

###### Akses halaman detail aplikasi yang telah diunduh.

###### Berikan rating dan ulasan untuk aplikasi tersebut.

###### Ulasan dan rating yang diberikan akan ditampilkan pada halaman detail aplikasi untuk dilihat oleh pengguna lain.

##### Pembelian In-Game Apps atau Money:

###### Login ke akun pengguna.

###### Akses halaman detail aplikasi yang ingin Anda beli in-game apps atau money-nya.

###### Pilih in-game apps atau money yang ingin Anda beli.

###### Pilih metode pembayaran yang Anda inginkan, seperti kartu kredit, debit, atau e-wallet.

###### Ikuti petunjuk pembayaran yang diberikan.

###### Setelah pembayaran berhasil, in-game apps atau money akan ditambahkan ke akun Anda.

## Hasil dan Pembahasan Inti

Setelah melakukan proses SDLC untuk merancang database PirtApps, maka kita sudah mendapat semua informasi yang dibutuhkan untuk membuat database bentuk physical, mulai dari penggambaran desain konseptual, logical, dan physical dilanjutkan dengan menganalisa kebutuhan fitur pengguna serta developer, maka kita akan lanjut kedalam koding physical design yaitu mySQL, pembahasan dimulai dari struktur database MySQL, dilanjutkan dengan Kueri yang terdapat di View dan Procedure.

### Struktur Database

A screenshot of a computer

Description automatically generated

Fig. 20 Struktur dari Database PirtApps

A screenshot of a computer

Description automatically generated

Fig. 21 Stuktur dari tabel developer

A screenshot of a computer

Description automatically generated

Fig. 22 Struktur dari tabel games

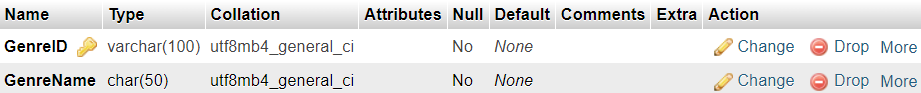


Fig. 23 Struktur dari tabel genre

A screenshot of a computer

Description automatically generated

Fig. 24 Struktur dari tabel payment

A screenshot of a computer

Description automatically generated

Fig. 25 Struktur dari tabel purchase

A screenshot of a computer

Description automatically generated

Fig. 26 Struktur dari tabel review

A screenshot of a computer

Description automatically generated

Fig. 27 Struktur dari tabel user

A screenshot of a computer

Description automatically generated

Fig. 28 Struktur dari tabel wishlist

### **Contoh** Kueri

View



Fig. 29 View Tabel Pembelian

CREATE VIEW purchase\_detail AS

SELECT

u.Username AS UserName,

g.Tittle AS GameTitle,

p.Amount,

p.PurchaseDate

FROM

purchases p

JOIN

user u ON p.UserID = u.UserID

JOIN

games g ON p.GameID = g.GameID

ORDER BY u.Username ASC;

A screenshot of a phone

Description automatically generated

Fig. 30 View tabel judul game

CREATE VIEW games\_list AS

SELECT

g.Tittle AS Title

FROM

games AS g

ORDER BY g.Tittle ASC;

A screenshot of a computer

Description automatically generated

Fig. 31 View Tabel detail developer

CREATE VIEW developers\_list AS

SELECT

d.DeveloperName,

d.Website,

d.Country

FROM

developers AS d

ORDER BY d.DeveloperName ASC;

### Diskusi

penulis mengalami beberapa tantangan selama proses penerjemahan model data logis ke implementasi fisik dengan MySQL seperti relasi antar tabel yang harus dihubungkan dalam keadaan kosong (tanpa data) dan kompleksitas model data karena berkaitan dengan pembelian dan pembayaran. Solusi yang diterapkan penulis untuk relasi antar tabel adalah dengan mengosongkan data tiap tabel dan menggunakan restrict untuk relasi antar tabel sehingga lebih aman untuk menyimpan banyak data yang saling berhubungan.

# BAB 5

**KESIMPULAN**



## Kesimpulan

SDLC (System Development Life Cycle) menjadi metode paling baik untuk merancang struktur database untuk platform aplikasi web PirtApps, dimulai dari berbagai tahapan hingga didapatkannya analisa untuk kebutuhan pengguna dan developer. Penggunaan SDLC yang tepat dapat membantu dalam **memenuhi kebutuhan pengguna dan developer dengan baik**, seperti kebutuhan akan pendaftaran akun yang mudah, penjelajahan aplikasi yang intuitif, pengunggahan aplikasi yang lancar, pengelolaan aplikasi yang efektif, komunikasi yang jelas dengan tim platform, pengunduhan aplikasi yang aman, sistem ulasan yang transparan, dan kemudahan pembelian in-game apps.

Prosedur dan fitur sangat dibutuhkan untuk memenuhi kebutuhan pengguna dan developer seperti pendaftaran akun, penjelajahan aplikasi, pengunggahan aplikasi, pengelolaan aplikasi, komunikasi dengan tim platform hingga pengunduhan, ulasan dan pembelian ingame apps. Dengan penggunaan SDLC dan analisa kebutuhan yang baik untuk pengguna dan developer, platform aplikasi PirtApps berpotensi menjadi platform yang handal dan bermanfaat bagi developer dan pengguna, sehinga bisa mendorong pertumbuhan ekosistem aplikasi yang inovatif dan dinamis.

## Saran

Penelitian selanjutnya dapat berfokus pada pengembangan Antarmuka Pengguna (UI) dan keamanan untuk memberikan pengalaman pengguna yang lebih nyaman dan aman. Hal ini dapat dilakukan dengan merancang UI yang lebih intuitif dan mudah digunakan, serta memperkuat sistem keamanan platform untuk melindungi data sensitif pengguna dan pengembang. Lalu mengembangkan fitur-fitur baru yang inovatif dan bermanfaat. Fitur-fitur baru ini dapat berupa komunitas online, sistem ulasan yang canggih, atau fitur-fitur lain yang dapat membantu meningkatkan pengalaman pengguna dan mendorong pertumbuhan ekosistem aplikasi.

# References

Anggriawan, F., & Leonardi, P. H. (2017). *PLATFORM BUILD-ME : Mempertemukan Stakeholder Bangunan Dengan Konsumen BUILD-ME PLATFORM : Meeting The Building Stakeholders With The Consumers*. <https://repository.unikom.ac.id/54575/1/vii-5-firman-anggriawan-platform-build-me-mempertemukan-stakeholder-bangunan-dengan-konsumen-halaman-genap.pdf>

Daud, D. E. (2023, February 9). *Software Development Life Cycle*. Cyber. <https://student-activity.binus.ac.id/csc/2023/02/software-development-life-cycle/>

Firman, A., Wowor, H. F., & Najoan, X. (2016). Sistem Informasi Perpustakaan Online Berbasis Web. *Jurnal Teknik Elektro Dan Komputer*, *5*(2), 29–36. <https://doi.org/10.35793/jtek.5.2.2016.11657>

GeeksforGeeks. (2023, November 24). *Use Case Diagram*. GeeksforGeeks. <https://www.geeksforgeeks.org/use-case-diagram/>

HIMASIS. (2022, March 28). *Mengenal ”Apa itu draw.io ?” Politeknik STMI Jakarta*. HIMASIS. <https://himasis.org/artikel/307-mengenal-apa-itu-draw-io>

Intern, D. (2021, May 4). *Apa itu Wireframe? Perbedaan Wireframe, Mockup, dan Prototype*. Dicoding Blog. <https://www.dicoding.com/blog/wireframe-adalah/>

Kustiyahningsih, Y., & Anamisa, D. R. (2011). Pemrograman basis data berbasis WEB menggunakan PHP & MySQL. *Yogyakarta: Graha Ilmu*, *20*.

mrasyidridho@student.telkomuniversity.ac.id. (2024, April 18). *Pengertian Flowchart: Fungsi, Jenis, Simbol, dan Contohnya*. S1 Teknik Elektro. <https://bee.telkomuniversity.ac.id/pengertian-flowchart-fungsi-jenis-simbol-dan-contohnya/>

Putra, D. W. T., & Andriani, R. (2019). Unified Modelling Language (UML) dalam Perancangan Sistem Informasi Permohonan Pembayaran Restitusi SPPD. *Jurnal TeknoIf*, *7*(1), 32. <https://doi.org/10.21063/jtif.2019.v7.1.32-39>

Rohi Abdulloh. (2016). *Easy & Simple - Web Programming*. Elex Media Komputindo.

Rohi Abdulloh. (2023). *7 in 1 Pemrograman Web untuk Pemula (Update Version)*. Elex Media Komputindo.

Sanjaya. (2015). *Materi Pembelajaran Pembuatan Aplikasi dan Web*. Yogyakarta: Amikom.

Susilawati, T., Yuliansyah, F., Romzi, M., & Aryani, R. (2020). MEMBANGUN WEBSITE TOKO ONLINE PEMPEK NTHREE MENGGUNAKAN PHP DAN MYSQL. *JTIM: Jurnal Teknik Informatika Mahakarya*, *03*(1), 35–44. <https://journal.unmaha.ac.id/index.php/jtim/article/download/19/19/134>

Team Biznet. (n.d.). *Mengenal Apa itu phpMyAdmin, Fungsi, Fitur, Hingga Cara Install | Biznet Gio*. Www.biznetgio.com. https://www.biznetgio.com/news/apa-itu-phpmyadmin