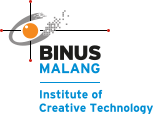
**Perancangan Database Sistem Penjualan Tiket Online Theater JKT48 Berbasis Website**

LAPORAN PROYEK AKHIR

MATA KULIAH ISYS6028 – DATABASE SYSTEMS

KELAS BC20



Oleh :

I Putu Krisna Dharma Saputra - 2301924353

Hafizh Naufal Hizbullah – 2301886310

Ida Bagus Rama Harischandra - 2301913141

Semester Ganjil 2020/2021

MALANG

**LEMBAR PERSETUJUAN PROYEK AKHIR**

**Perancangan Database Sistem Penjualan Tiket Online Theater JKT48 Berbasis Website**

MATA KULIAH ISYS6028 – DATABASE SYSTEMS

KELAS BC20

Semester Ganjil 2020/2021

Laporan akhir proyek ini adalah benar karya kami:

**(I Putu Krisna Dharma S.)** **(Hafizh Naufal Hizbullah)**  **(Ida Bagus Rama H.)**

**2301924353 2301886310 2301913141**

**Malang, 20 Januari 2021**

**(Sidharta)**

**D5848**

**DAFTAR ISI**

**COVER**

**LEMBAR PERSETUJUAN i**

**DAFTAR ISI ii**

**DAFTAR GAMBAR iv**

**DAFTAR TABEL v**

**BAB I LATAR BELAKANG**

* 1. **Latar Belakang 1**
  2. **Rumusan Masalah 1**
  3. **Tujuan Penelitian 1**
  4. **Manfaat Penelitian 2**
  5. **Sistematika Penulisan 2**

**BAB II KAJIAN PUSTAKA**

* 1. **Database 4**
     1. **Pengertian Database 4**
     2. **DBMS 4**
  2. **Website 5**
  3. **UML (Unified Modeling Language) 5**
  4. **ERD (Entity Relationship Diagram) 6**
     1. **Pengertian ERD 6**
     2. **Kardinalitas Relasi 6**

**BAB III ANALISIS**

* 1. **System Definition 7**
  2. **Identifikasi Permasalahan 9**
  3. **Identifikasi Kebutuhan Pengguna 9**

**BAB IV DATABASE DESIGN**

* 1. **Conceptual Design 10**
     1. **Entity Types 10**
     2. **Relationship Types 11**
     3. **List of Attributes 12**
     4. **List of Attribute Domains 14**
     5. **Candidate Keys, Primary Keys, and Alternate Keys 15**
     6. **Entity Relationship Diagram 16**
  2. **Logical Model (Class Diagram) 17**
  3. **Physical Model 18**
     1. **Analyze Transaction 18**
     2. **User View, Procedure, and Function 18**
  4. **Design Sistem WEBSITE(Mockup) 21**

**DAFTAR PUSTAKA 25**

**LAMPIRAN 26**

**PENILAIAN (ASSESSMENT) 38**

**DAFTAR GAMBAR**

**Gambar 3.1 *Use Case Diagram* Website jkt48.com7**

**Gambar 3.2 *Flow Chart* Proses Pemesanan Tiket 8**

**Gambar 4.1 Entity Relationship Diagram 16**

**Gambar 4.2 Logical Model 17**

**Gambar 4.3 Tabel tb\_fans 18**

**Gambar 4.4 Tabel tb\_staff 19**

**Gambar 4.5 Tabel tb\_member 19**

**Gambar 4.6 Tabel tb\_team 19**

**Gambar 4.7 Tabel tb\_setlist 19**

**Gambar 4.8 Tabel tb\_detail\_transaction 20**

**Gambar 4.9 Tabel tb\_ticket 20**

**Gambar 4.10 Tabel tb\_schedule 20**

**Gambar 4.11 Tampilan Awal Website 21**

**Gambar 4.12 Menu Login (User/Fans) 21**

**Gambar 4.13 Menu Register Tahap 1 22**

**Gambar 4.14 Menu Register Tahap 2 22**

**Gambar 4.15 Menu Register Tahap 3 22**

**Gambar 4.16 Menu Profile 23**

**Gambar 4.17 Menu Login (Admin/Staff) 23**

**Gambar 4.18 Menu Graphic Perkembangan Penjualan Tiket 24**

**DAFTAR TABEL**

**Tabel 4.1 Entity Types 10**

**Tabel 4.2 Relationship Types 11**

**Tabel 4.3 Attributes 12**

**Tabel 4.4 Candidate keys, Primary Keys, and Alternate Keys 15**

**Tabel 4.5 Analyze Transaction 18**

**BAB I**

**LATAR BELAKANG**

**1.1 Latar belakang**

JKT48 merupakan *subgrup* dari *idol grup* ternama di Jepang AKB48 yang diproduseri oleh Yasushi Akimoto. Nama JKT48 sendiri berasal dari kota di Indonesia yaitu Jakarta dan sejarah angka 48 merujuk pada nama presiden group AKB48 Shiba Kotaro (芝幸太郎) dimana *Shiba* artinya 48. JKT48 didirikan pada 11 September 2011 di sebuah acara AKB48 yang bertempat di daerah Makuhari Messe, Chiba, Jepang [1]. Generasi pertama yang diperkenalkan pada 3 November 2011 terdapat 28 anggota. Per 26 Desember 2020, JKT48 memiliki 48 orang anggota tim secara individu, termasuk 1 orang yang merangkap di tiga tim.

JKT48 mengadakan pertunjukan rutin hampir setiap hari di teater JKT48, lantai 4 mal fX Sudirman, Jakarta. Pemesanan tiket teater JKT48 masih dilakukan secara manual yaitu bagian ticketing akan mencatat pesanan tiket teater dari para fans di loket tiket teater JKT48, lantai 4 mal fX. Namun pembelian tiket secara offline ini memunculkan resiko terjadinya masalah. Masalah tersebut yaitu pada saat pencatatan nama dan nomor identitas fans pada guest card. Pada saat proses pemesanan tiket teater JKT48, bagian ticketing akan mencatat nama pelanggan yang melakukan pemesanan tiket teater JKT48 sesuai dengan nama yang tertera pada identitas diri fans seperti KTP. Bagian ticketing tidak boleh melakukan kesalahan dalam pencatatan nama tersebut. Begitu juga pada saat mencatat nomor identitas diri yang digunakan seperti nomor KTP. Selain itu, para fans juga harus mengantri sangat Panjang untuk memesan tiket ini. Hal-hal ini membuat pemesanan tiket secara offline menjadi kurang efektif dan efisien.

Dari permasalahan yang ditemukan dari proses pembelian tiket tersebut, kami menemukan sebuah solusi yaitu dengan memanfaatkan adanya *website*. Melalui pembuatan sebuah *website* untuk pembelian tiket, para fans JKT48 dapat lebih efektif dan efisien dalam melakukan transaksi. Peningkatan keefektifan dan keefisienan penjualan tiket oleh sebuah *website* dapat dilihat dari banyaknya fitur yang dapat diimplementasikan ke dalam *website* mulai dari pemesanan tiket teater JKT48secara *online,* melihat jadwal show JKT48 secara *online,* hingga melihat history pembelian tiket teater JKT48 secara online.

**1.2 Rumusan Masalah**

Berdasarkan paparan latar belakang di atas, maka dapat dirumuskan beberapa permasalahan yang ingin dicari solusinya.

1. Bagaimana cara membeli tiket teater JKT48 dengan lebih efektif dan efisien?

2. Bagaimana implementasi penjualan tiket teater JKT48 menggunakan website yang terhubung dengan database?

**1.3 Tujuan Penelitian**

Adapun tujuan penulisan yang ingin dicapai dalam penyusunan laporan ini yaitu sebagai berikut.

1. Mengetahui cara membeli tiket teater JKT48 dengan lebih efektif dan efisien.

2. Mengetahui implementasi penjualan tiket teater JKT48 menggunakan website yang terhubung dengan database.

**1.4 Manfaat Penelitian**

Manfaat yang diharapkan mampu disumbangkan dalam laporan ini adalah sebagai berikut.

1. Manfaat teoritis, dapat mengkontribusi pada penambahan teori dan temuan mengenai perancangan database untuk sistem penjualan tiket online teater JKT48 berbasis website.

2. Manfaat praktis yakni sebagai berikut.

1) Bagi penulis, menambah wawasan mengenai perancangan database untuk sistem penjualan tiket online teater JKT48 berbasis website.

2) Bagi fans JKT48, website ini dapat memudahkan pembelian tiket dengan lebih efektif dan efisien.

**1.5 Sistematika Penulisan**

COVER

*Cover* berisi halaman pembuka dan sampul dari penelitian.

LEMBAR PERSETUJUAN

Lembar Persetujuan berisi persetujuan dari dosen dan mahasiswa untuk menjalankan penelitian ini.

DAFTAR ISI

Daftar isi berisi keseluruhan konten-konten dari hasil penelitian yang telah dilakukan.

BAB I LATAR BELAKANG

Pada bab 1 dijelaskan latar belakang akan pembuatan artikel ini, rumusan masalah, tujuan, dan manfaat dari pembuatan artikel ini

BAB II KAJIAN PUSTAKA

Tinjauan pustaka berisi materi-materi dan dasar-dasar teori yang digunakan sebagai landasan pembuatan sistem.

BAB III ANALISIS

*Analysis* akan menjelaskan mengenai definisi sistem menggunakan *use case diagram*, gambaran umum sistem, dan identifikasi kebutuhan pengguna.

BAB IV DESIGN

Bab *design* berisi mengenai model konseptual dari sistem, model logis, dan model fisik dari sistem yang akan dibuat

DAFTAR PUSTAKA

Daftar pustaka berisi sumber-sumber artikel, buku, maupun jurnal yang digunakan sebagai referensi dalam pembuatan artikel.

LAMPIRAN

Lampiran berisi *screenshot* dari website dan query-query *function, view, dan procedure* dalam SQL server.

PENILAIAN (ASSESSMENT)

Halaman penilaian berisi tabel-tabel pembagian persentase kerja dan penilaian per mahasiswa

**BAB II**

**KAJIAN PUSTAKA**

**2.1 Database**

**2.1.1 Pengertian Database**

Database adalah kumpulan informasi yang disusun dan merupakan suatu kesatuan yang utuh yang disimpan di dalam perangkat keras (komputer) secara sistematis sehingga dapat diolah menggunakan perangkat lunak [2].

**2.1.2 DBMS**

Untuk mengelola suatu database diperlukan suatu software yang disebut DBMS. DBMS (*Database Management System)* adalah perangkat lunak untuk mengendalikan pembuatan pemeliharaan, pengolahan, dan penggunaan database dalam skala yang besar [3]. DBMS juga dirancang untuk memudahkan memanipulasi data. DBMS sudah menjadi peran atau kunci utama serta bagian standar di bagian pendukung sebuah perusahaan.

Pada penelitian ini kami menggunakan DBMS Microsoft SQL Server. Microsoft SQL Server merupakan produk RDBMS (Relational Database Management Sistem) yang dibuat oleh Microsoft [4]. Secara garis besar, SQL Server mempunyai 3 (Tiga) jenis Transact SQL yaitu :

1. Data Definition Language (DDL), merupakan bagian dari sistem manajemen database yang dipakai untuk mendefinisikan dan mengatur semua atribut dan properti dari sebuah database.

2. Data Manipulation Language (DML), merupakan perintah – perintah yang digunakan untuk menampilkan, menambah, mengubah, dan menghapus data di dalam Objek – Objek yang didefinisikan oleh DDL.

3. Control Language (DCL), digunakan untuk mengontrol hak – hak pada Objek – Objek database.

Berikut beberapa query yang kami gunakan dalam Menyusun databaseticketing JKT48:

# **-** Create Query

CREATE berguna untuk membuat suatu table, view, ataupun store procedure. Adapun contoh Query CREATE adalah sebagai berikut :

CREATE TABLE tb\_setlist

(

       SetlistId CHAR (5) NOT NULL PRIMARY KEY,

       SetlistName VARCHAR (20) NOT NULL,

TeamId CHAR (5) NOT NULL FOREIGN KEY REFERENCES tb\_team (TeamId) ON UPDATE CASCADE ON DELETE CASCADE,

       CONSTRAINT Check\_SetlistId CHECK (SetlistId LIKE ('SL[0-9][0-9][0-9]'))

);

-       Insert Query

INSERT berguna untuk memasukkan atau menyimpan data baru ke dalam database. Adapun contoh Query INSERT adalah sebagai berikut :

INSERT INTO tb\_setlist (SetlistId, SetlistName, TeamId)

VALUES ('SL001', 'High Tension', 'TM004'),

              ('SL002', 'Saka Agari', 'TM001'),

              ('SL003', 'Rapsodi', 'TM002'),

              ('SL004', 'Idol no Yoake', 'TM003');

-       Update Query

UPDATE berguna untuk memperbarui data yang ada dalam database. Adapun contoh Query UPDATE adalah sebagai berikut ini:

UPDATE MsCustomer

SET CustomerName = CONCAT('Ms.',CustomerName)

WHERE CustomerId in('CU002','CU003');

-       Count Query COUNT berguna untuk menghitung jumlah record pada table tertentu dalam database.

SELECT COUNT(1) [Sales Count]

FROM tb\_header\_transaction

**2.2 Website**

Pada dasarnya website adalah kepanjangan dari *World Wide Web* (WWW). Informasi WWW ini disimpan pada web server untuk dapat diakses dari jaringan browser terlebih dahulu, seperti Internet Explorer atau Mozilla Firefox [5]. Sebuah web page adalah dokumen yang ditulis dalam format HTML (Hyper Text Markup Language), yang hampir selalu bisa diakses melalui HTTP, yaitu protokol yang menyampaikan informasi dari server website untuk ditampilkan kepada para pemakai melalui web browser baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (hyperlink).

Bersifat statis apabila isi informasi website tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik website. Bersifat dinamis apabila isi informasi website selalu berubah-ubah, dan isi informasinya interaktif dua arah berasal dari pemilik serta pengguna website. Contoh website statis adalah berisi profil perusahaan, sedangkan website dinamis adalah seperti *Friendster*, *Multiply*, dll. Dalam sisi pengembangannya, website statis hanya bisa diupdate oleh pemiliknya saja, sedangkan website dinamis bisa diupdate oleh pengguna maupun pemilik [7].

**2.3 UML (Unified Modeling Language)**

UML (Unified Modeling Language) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek [6]. UML mempunyai beberapa atau sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Diagram tersebut akan menggambarkan atau mendokumentasikan beberapa aspek dari sebuah sistem.

Pada penelitian ini kami menggunakan salah satu dari UML yaitu Use Case Diagram. Use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

**2.4 ERD (Entity Relationship Diagram)**

**2.4.1 Pengertian ERD**

ERD (Entity Relationship Diagram) adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model [5]. Di Dalam hubungan ini tersebut dinyatakan yang utama dari ERD adalah menunjukan objek data (Entity) dan hubungan (Relationship), yang ada pada Entity berikutnya.

Simbol-simbol dalam ERD (Entity Relationship Diagram) adalah sebagai berikut:

a. Entitas: suatu yang nyata atau abstrak yang mempunyai karakteristik dimana kita akan menyimpan data.

b. Atribut: ciri umum semua atau sebagian besar instansi pada entitas tertentu.

c. Relasi: hubungan alamiah yang terjadi antara satu atau lebih entitas.

d. Link: garis penghubung atribut dengan kumpulan entitas dan kumpulan entitas dengan relasi.

**2.4.2. Kardinalitas Relasi**

a. Satu ke satu (One to One) Setiap elemen dari Entitas A berhubungan paling banyak dengan elemen pada Entitas B. Demikian juga sebaliknya setiap elemen B berhubungan paling banyak satu elemen pada Entitas A.

b. Satu ke banyak (One to Many) Setiap elemen dari Entitas A berhubungan dengan maksimal banyak elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B berhubungan dengan paling banyak satu elemen di Entitas A.

c. Banyak ke satu (Many to One) Setiap elemen dari Entitas A berhubungan paling banyak dengan satu elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B behubungan dengan maksimal banyak elemen di entitas A.

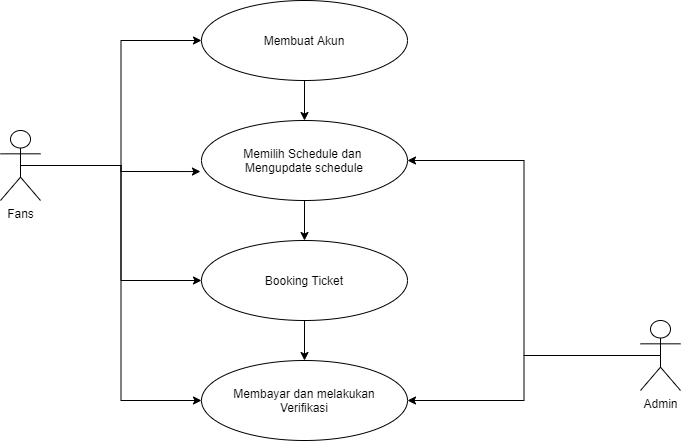
d. Banyak ke banyak (Many to Many) Setiap elemen dari Entitas A berhubungan maksimal banyak elemen pada Entitas B demikian sebaliknya.

**BAB 3**

**ANALISIS**

**3.1. System Definition**

Berikut merupakan Use Case dari aplikasi yang telah kami buat

****

**Gambar 3.1 *Use Case Diagram* Website jkt48.com**

Dalam gambar tersebut terdapat 2 aktor yang terlibat :

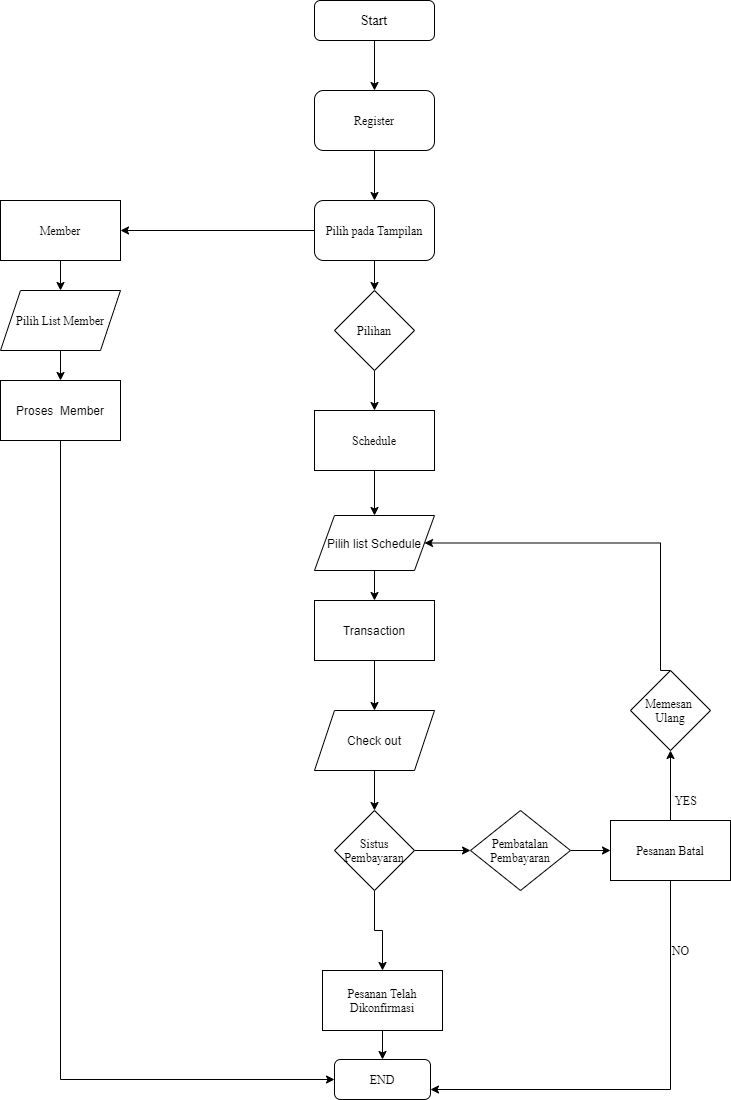
1. *Fans*

Aktor ini merupakan pengguna dalam Web ini dan dapat melakukan berbagai aktivitas seperti memilih fitur, memilih schedule untuk melihat update terbaru, melakukan pembayaran serta verifikasi pembayaran..

1. *Admin*

Aktor ini merupakan *admin* yang hanya bertugas untuk mengawasi transaksi pada customer serta memverifikasi pemesanan yang telah dilakukan oleh *Fans*

Berikut dibawah merupakan *flowchart* bagaimana skenario ketika pengguna melakukan aktivitas pada aplikasi kami.

Pada *flowchart* tersebut ada beberapa alur diantaranya, ketika pengguna memilih tampilan list schedule yang dimana user akan memilih tanggal dan waktu yang sudah tertera pada list schedule yang sudah disediakan. Setelah itu user akan memverifikasikan pembayaran sesuai dengan pilihan, jika user menolak maka transaksi sebelumnya akan dibatalkan dan user dapat memesan ulang sesuai schedule yang diinginkan.

**Gambar 3.2 *Flow Chart* Proses Pemesanan Tiket**

* 1. **Identifikasi Permasalahan**

Berdasarkan latar belakang yang dijelaskan pada bab 1, permasalahan yang berhasil ditemukan adalah:

* Terdapat resiko terjadinya kesalahan pengisian data saat pemesanan tiket yang dilakukan secara manual.
* *Fans* kurang mengetahui *show* yang akan berlangsung.
* Antrian panjang yang harus dilalui oleh *fans* saat ingin memesan tiket.
* Sistem pemesanan tiket di loket secara offline kurang efektif dan efisien.
* Susahnya mendapatkan informasi mengenai show yang akan dibatalkan/*delay*.
  1. **Identifikasi Kebutuhan Pengguna**

Hingga saat ini, fans dari JKT48 sudah sangat banyak di Indonesia. Dengan semakin berkembangnya teknologi, muncul lah website yang dapat memudahkan suatu kegiatan seperti mencari informasi, mencari hiburan, hingga melakukan transaksi. Pada kasus ini, kami sedang meneliti bagaimana suatu website dapat mempermudah transaksi pemesanan tiket yang biasanya dilakukan secara on site.

Sistem yang dibuat pada proyek/penelitian ini adalah sebuah *website* yang dihubungkan dengan *database* untuk mempermudah pembelian tiket tersebut. Melalui sistem ini, para fans JKT48 dapat lebih efektif dan efisien dalam melakukan transaksi. Peningkatan keefektifan dan keefisienan penjualan tiket oleh sebuah *website* dapat dilihat dari banyaknya fitur yang dapat diimplementasikan ke dalam *website* mulai dari pemesanan tiket teater JKT48secara *online,* melihat jadwal show JKT48 secara *online,* hingga melihat history pembelian tiket teater JKT48 secara online.

**BAB 4**

**DATABASE DESIGN**

**4.1 Conceptual Design**

**4.1.1 Entity Types**

*Entity types* adalah kumpulan dari entitas yang memiliki atribut serupa. Pada setiap tabel, kita memiliki setiap baris sebagai entitas dan mereka memiliki atribut yang sama. Berikut Tabel Entity types dari database kami.

|  |  |  |  |
| --- | --- | --- | --- |
| Entity Name | Description | Aliases | Occurance |
| tb\_team | Berisikan Nama JKT yang terbagi menjadi 4 Team J,K3,T,Academy A | - | Setiap anggota dari JKT akan dikelompokan kedalam 1 team yang ada |
| tb\_staff | Berisikan data diri staff yang bertugas | - | Setiap staff akan miliki role perkerjaan dan waktu yang berbeda |
| tb\_member | Berisikan data diri dari setiap anggota yang terdaftar dalam JKT | - | Setiap member akan dibagi menjadi teamnya masing-masing |
| tb\_fans | Berisi data diri pengemar dari member JKT | - | Setiap penggemar JKT bisa menonton member sesuai dengan jadwal dan kegemarannya |
| tb\_shedule | Berisikan jadwal tayang, duarasi, tanggal pelakasanaan | - | Setiap jadwal memiliki jadwak tanyang waktu dan tempat yang berbeda |
| tb\_header\_transcation | Berisi jadwal waktu dan tempat yang ditentukan | - | Jadwal waktu dan tempat sudah ditetapkan secara spesifik dan detail |
| tb\_ticket | Berisi jenis tiket yang sudah dibeli oleh fans sesuai dengan jadwal yang ditentukan | - | Setiap satu ticket menjelaskan tentang jadwal, waktu dan tempat, dan juga no ticket yang sudah dipesan sesuai jadwal tayang yang ada |
| tb\_setlist | Berisi jenis setlist yang akan ditampilkan oleh team yang telah ditentukan | - | Setiap jenis setllist aka ditampilkan sesuai dengan team yang telah ditentukan |

**Tabel 4.1 Entity Types**

**4.1.2 Relationship Types**

*Relationship Types* adalah kumpulan antar dari tipe entiti yang dimana hubungan yang diidentifikasikan secara unik, yang dari setiap entiti yang ada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entity Name | Mutiplicity | Relationship | Multiplicity | Entity Name |
| Fans | 1..1 | Memiliki | 1..1 | FansId |
|  | 1..1 | Memiliki | 1..1 | TeamId |
| tb\_fans | 1..1 | Berkaitan | 1..1 | Fans |
|  | 1..1 | Berkaitan | 1..1 | Member |
| tb\_team | 1..1 | Berkaitan | 1..1 | Fans |
|  | 1..1 | Berkaitan | 1..1 | Team Member |
| Member | 1..1 | Memiliki | 1..1 | MemberId |
| tb\_member | 1..1 | Berkaitan | 1..1 | Member |
|  | 1..1 | Berkaitan | 1..1 | Trans |
| Trans | 1..1 | Memiliki | 1..1 | TransId |
|  | 1..1 | Memiliki | 1..1 | ScheduleId |
| tb\_trans | 1..1 | Berkaitan | 1..1 | Transcation |
|  | 1..1 | Berkaitan | 1..1 | Ticket |
| tb\_schedule | 1..1 | Berkaitan | 1..1 | Transcation |
|  | 1..1 | Berkaitan | 1..1 | Schedule |
| Ticket | 1..1 | Memiliki | 1..1 | TransId |
|  | 1..1 | Memiliki | 1..1 | TicketId |
| tb\_ticket | 1..1 | Berkaitan | 1..1 | Ticket |
|  | 1..1 | Berkaitan | 1..1 | Ticket |
| Staff | 1..1 | Memiliki | 1..1 | StaffId |
| tb\_staff | 1..1 | Berkaitan | 1..1 | Staff |
|  | 1..1 | Berkaitan | 1..1 | Member |

**Tabel 4.2 Relationship Type**

**4.1.3 Attributes**

*Attributes* adalah sebuah properti dari *entity* atau *relationship type.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entity Name | Atributte | Description | Data Type & Lenght | Nulls | Multi Valued |
| tb\_staff | StaffId | Identifikasi Staff | 5 Character | No | No |
| StaffName | Nama Staff | 50 Variable Character | No | No |
| StaffPosition | Jabatan Staff | 20 Variable Character | No | No |
| StaffSalary | Pendapatan Staff | Integer | No | No |
| tb\_fans | FansId | Identifikasi Fans | 5 Character | No | No |
|  | FansName | Nama Fans | 50 Variable Character | No | No |
|  | FansGender | Jenis Kelamin Fans | 10 Variable Character | No | No |
|  | FansBirthday | Tangal Lahir Fans | Date | No | No |
|  | FansAttend | Fans Mengahadiri | Interger | No | No |
|  | FansEmail | Alamat Email Fans | 20 Variable Character | No | No |
|  | FansPassword | Password Fans | 50 Variabel Character | No | No |
| tb\_member | MemberId | Identifikasi Member | 5 Character | No | No |
|  | MemberName | Nama Member | 50 Variable Character | No | No |
|  | MemberBirthday | Tanggal Lahir Member | Date | No | No |
|  | MemberCreated | Karir Member | Date | No | No |
|  | MemberPopularity | Popularty Member | Interger | No | No |
| tb\_team | TeamId | Identifikasi Team | 5 Character | No | No |
|  | TeamName | Nama Team | 10 Variable Character | No | No |
| tb\_setlist | SetlistId | Identifikasi Setlist | 5 Character | No | No |
|  | SetlistName | Nama Setlist | 20 Variable Character | No | No |
| tb\_schedule | ScheduleId | Identifikasi Jadwal | 5 Character | No | No |
|  | DateHeld | Tanggal Tayang | Date | No | No |
|  | TimeHeld | Waktu Tayang | 15 Character | No | No |
|  | ScheduleLimit | Limit jadwal | 1 Character | No | No |
|  | ScheduleShow | Jadwal Tayang | Interger | No | No |
| tb\_header\_transcation | TransId | Indetifikasi Transaksi | 5 Character | No | No |
|  | FansId | Indetifikasi Fans | 5 Character | No | No |
|  | TransDate | Tanggal Transaksi | Date | No | No |
|  | TransType | Jenis Transaksi | 10 Character | No | No |
| tb\_detail\_transcation | TransId | Identifikasi Transaksi | 5 Character | No | No |
|  | ScheduleId | Identifikasi Jadwal | 5 Character | No | No |
| tb\_ticket | TicketId | Identifikasi Ticket | 5 Character | No | No |
|  | TicketType | Jenis Ticket | 10 Character | No | No |
|  | TicketNo | Nomor Ticket | Interger | No | No |
|  | TransId | Identifikasi Transaksi | 5 Character | No | No |

**Tabel 4.3 Attributes**

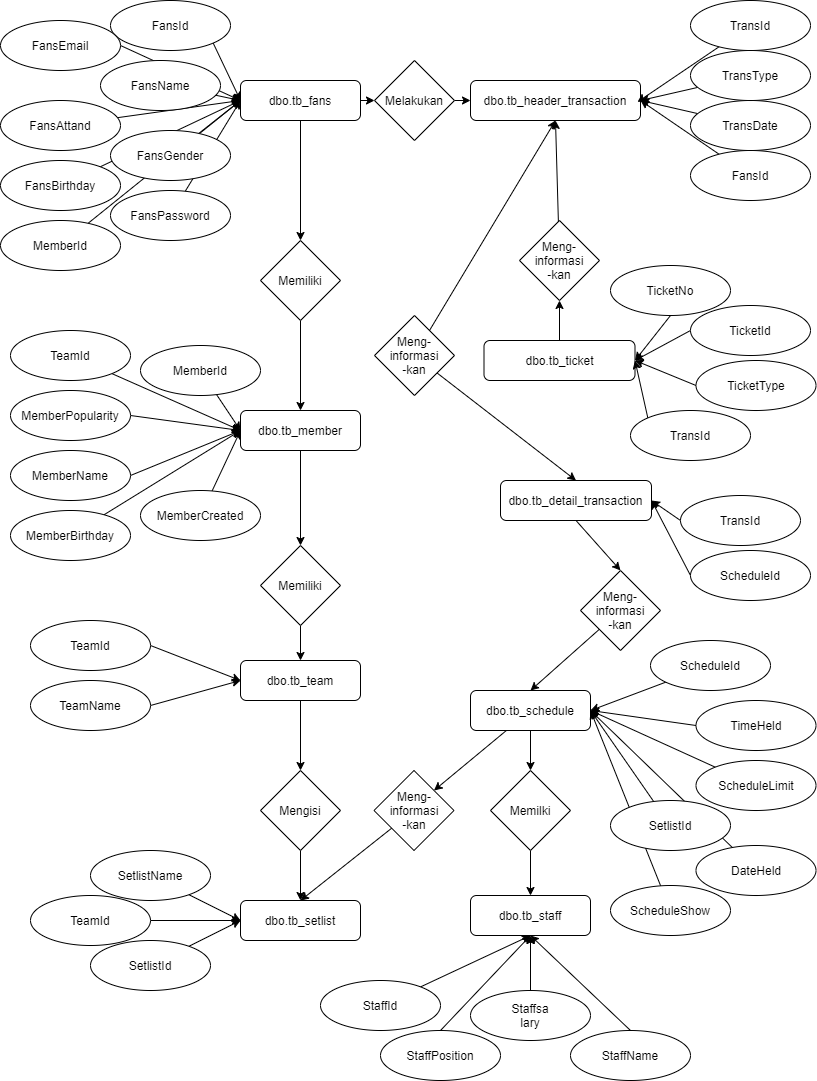
* + 1. **Attribute Domains**
* Attribut domain pada tabel FANS yang memenuhi FansId adalah constraint yang bernama FJ01 dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf  “FJ” dan karakter selanjutnya berupa angka dari 0 hingga 9999.
* Attribut domain pada tabel MEMBER yang memenuhi MemberId adalah constraint yang bernama MJ01 dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “MJ” dan karakter selanjutnya berupa angka dari 0 hingga 9999.
* Attribut domain pada tabel TEAM yang memenuhi TeamId adalah constraint yang bernama TM01 dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “TM” dan karakter selanjutnya berupa angka dari 0 hingga 9999.
* Attribut domain pada tabel SCHEDULE yang memenuhi ScheduleId adalah constraint yang bernama PRD2 dimana constraint ini menghubungkan PDId yang berada pada tabel COACH dengan PDId yang ada pada tabel PROGRAM DIET.
* Attribut domain pada tabel HEADER TRANSACTION yang memenuhi TransId adalah constraint yang bernamaHT01 dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “HT”dan karakter selanjutnya berupa angka dari 0 hingga 9999.
* Attribut domain pada tabel DETAIL TRANSACATION yang memenuhi TransId adalah constraint yang bernama TR01 dimana constraint ini menghubungkan  yang berada pada tabel DETAIl TRANSCATION dengan  yang ada pada tabel HEADER TRANSCATION.
* Attribut domain pada tabel TICKET yang memenuhi TicketId adalah constraint yang bernama TJ01 dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “TJ” dan karakter selanjutnya berupa angka dari 0 hingga 9999.

**4.1.5 Candidate Keys, Primary Keys and Alternate Keys**

*Keys* adalah Model dari *entity* dan *relationship* yang memiliki beberapa key. diantaranya mirip dengan Model relasional.

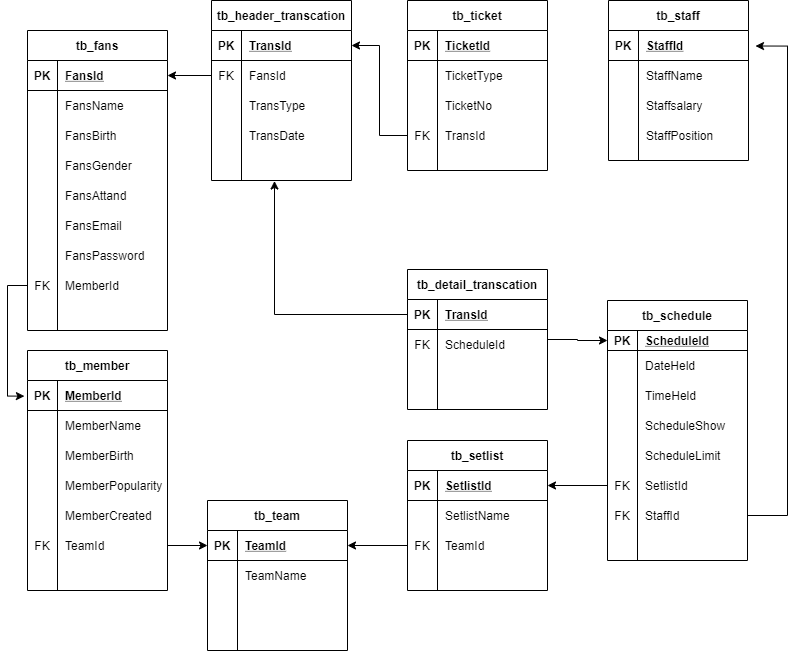
|  |  |  |  |
| --- | --- | --- | --- |
| Entity Name | Candidate Key | Primary Key | Alternate Key |
| dbo.fans | FansEmail | FansId | FansEmail  FansId |
| dbo.staff | StaffName | StaffId | StaffName  StaffId |
| dbo.member | MemberName | MemberId | MemberName |
| dbo.team | TeamName  SetlistId | TeamId | SetlistId |
| dbo.schedule | SetlistId | ScheduleId | ScheduleId  SetlistId |
| dbo.detailtransacation | ScheduleId | TransId | TransId  SchedueId |
| dbo.headertransaction | FansId | TransId | FansId |
| dbo.ticket | TransId | TicketId | TickeId  TransId |

**Tabel 4.4 Candidate, Primary dan Alternate Keys**

**4.1.6 Entity Relationship Diagram**

**Gambar 4.1 Entity Relationship Diagram**

* 1. **Logical Model (Class Diagram)**

****

**Gambar 4.2 Logical Model**

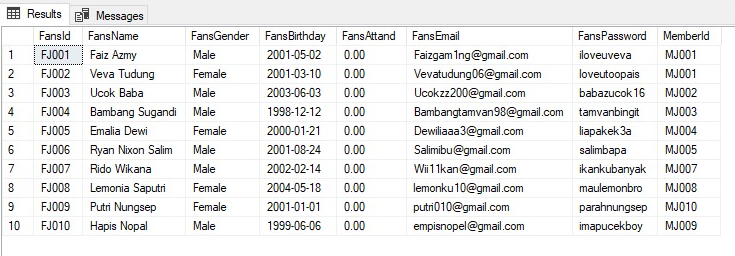
**4.3 Physical Model**

**4.3.1 Analyze Transaction**

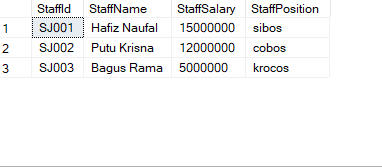
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Transaction/Relation** | **Fans** | | | | **Member** | | | | **Staff** | | | |
|  | **Read** | **Update** | **Delete** | **Insert** | **Read** | **Update** | **Delete** | **Insert** | **Read** | **Update** | **Delete** | **Insert** |
| **tb\_team** | **x** | **x** |  | **x** | **x** |  |  |  | **x** |  |  |  |
| **tb\_fans** | **x** |  |  |  | **x** | **x** |  |  | **x** | **x** | **x** | **x** |
| **tb\_member** | **x** | **x** |  | **x** | **x** |  |  |  | **x** | **x** | **x** | **x** |
| **tb\_staff** | **x** | **x** |  | **x** | **x** |  |  |  | **x** | **x** | **x** | **x** |
| **tb\_setlist** | **x** |  |  |  |  |  |  |  | **x** | **x** |  |  |
| **tb\_schedule** | **x** |  |  |  | **x** |  |  |  | **x** | **x** | **x** | **x** |
| **tb\_header\_transcation** | **x** | **x** | **x** | **x** | **x** |  |  |  | **x** |  | **x** | **x** |
| **tb\_detail\_transcation** | **x** | **x** | **x** | **x** | **x** |  |  |  | **x** |  |  |  |
| **tb\_ticket** | **x** |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** |

**Tabel 4.5 Analyze Transaction**

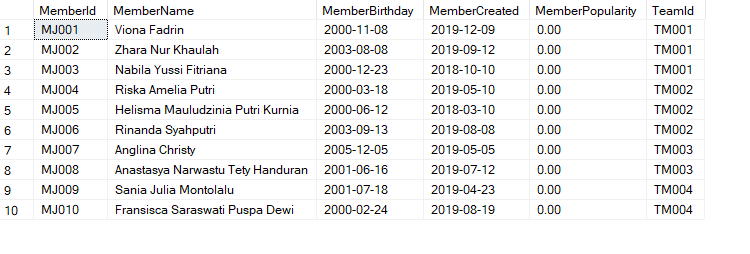
**4.3.2 User View, Procedure and Function**

****

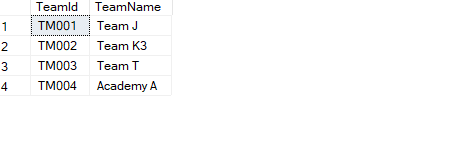
**Gambar 4.3 Tabel tb\_fans**

****

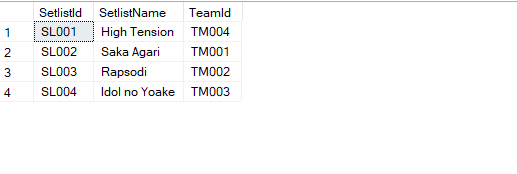
**Gambar 4.4 Tabel tb\_staff**

****

**Gambar 4.5 Tabel tb\_member**

****

**Gambar 4.6 Tabel tb\_team**

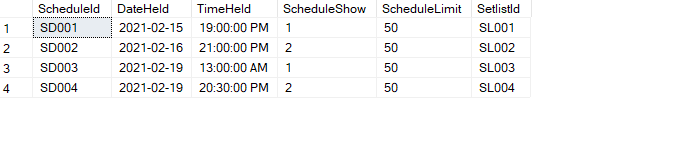
****

**Gambar 4.7 Table tb\_setlist**

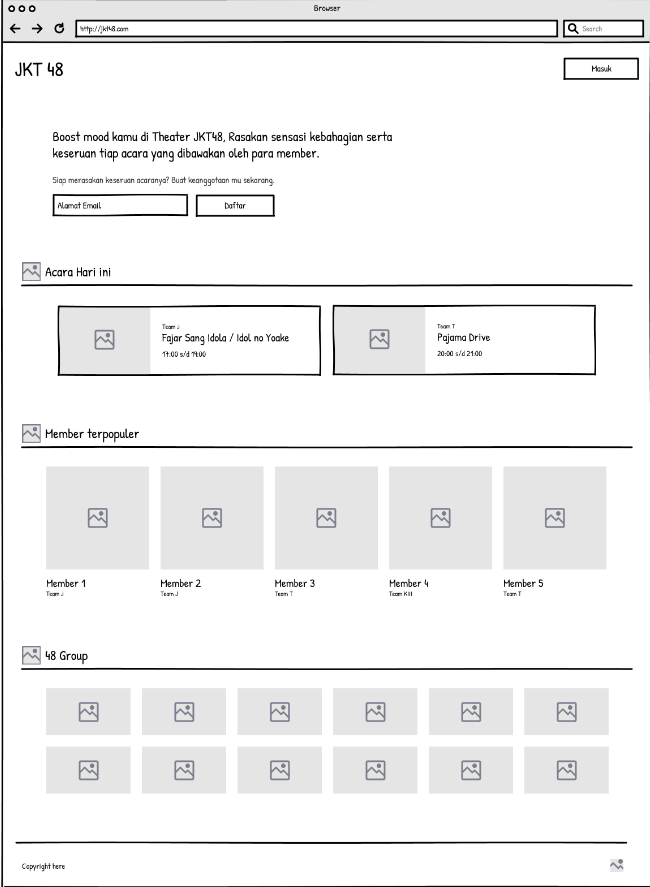
**Gambar 4.8 Table tb\_detail\_transaction**

****

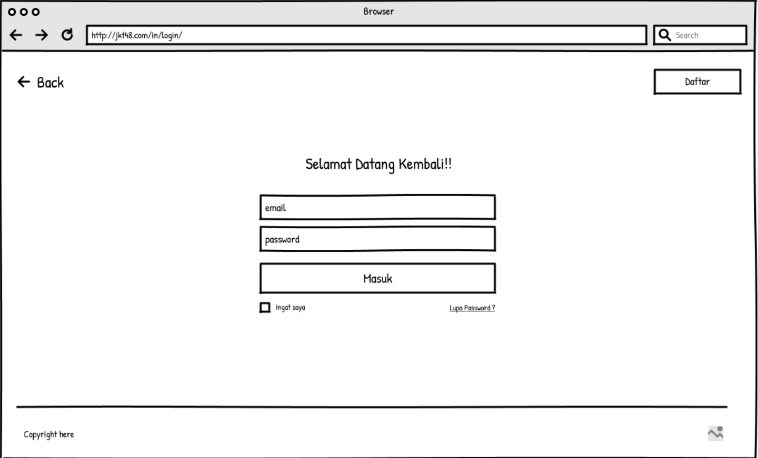
**Gambar 4.9 Table tb\_ticket**

****

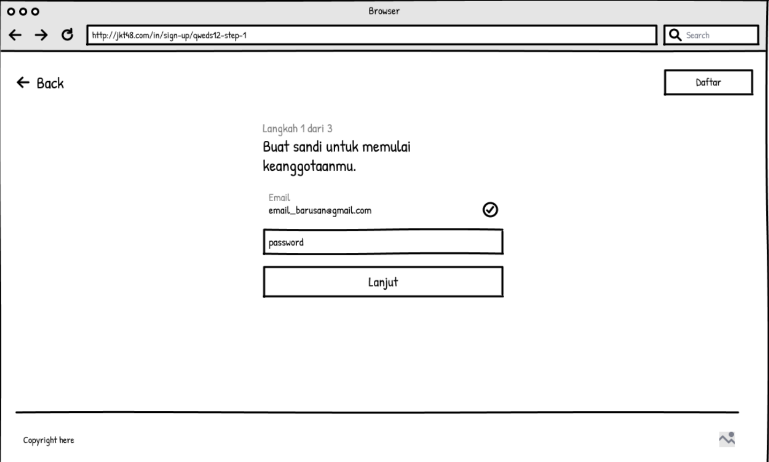
**Gambar 4.10 Table tb\_schedule**

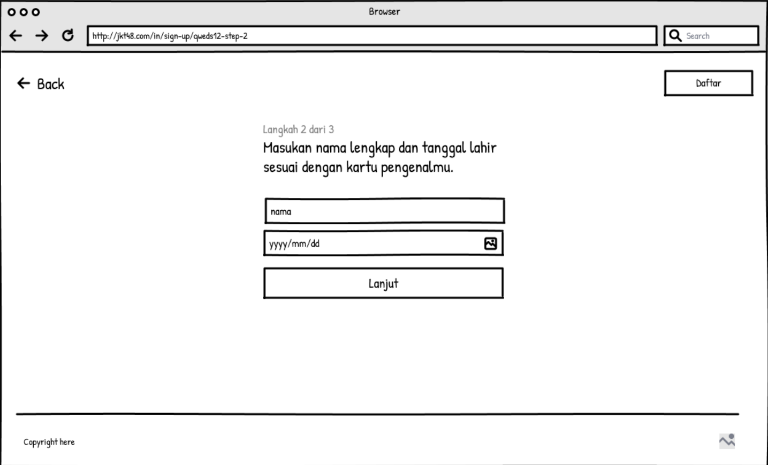
* 1. **Design Sistem WEBSITE** **(Mockup)**

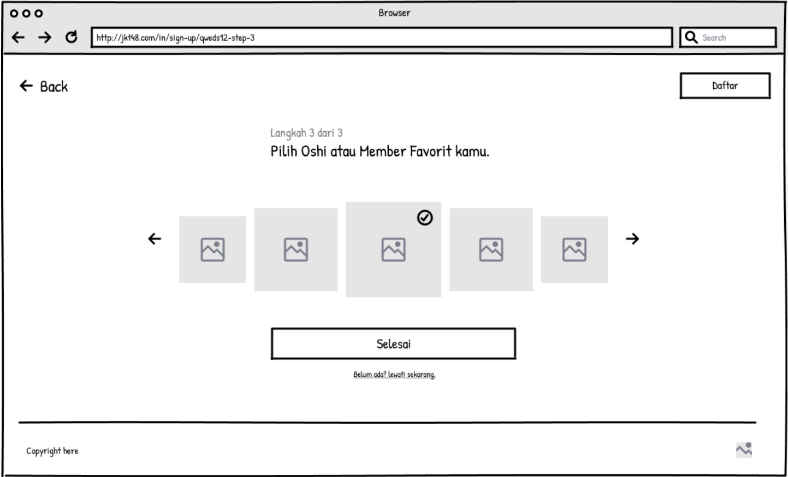
**Gambar 4.11 Tampilan Awal Website**

****

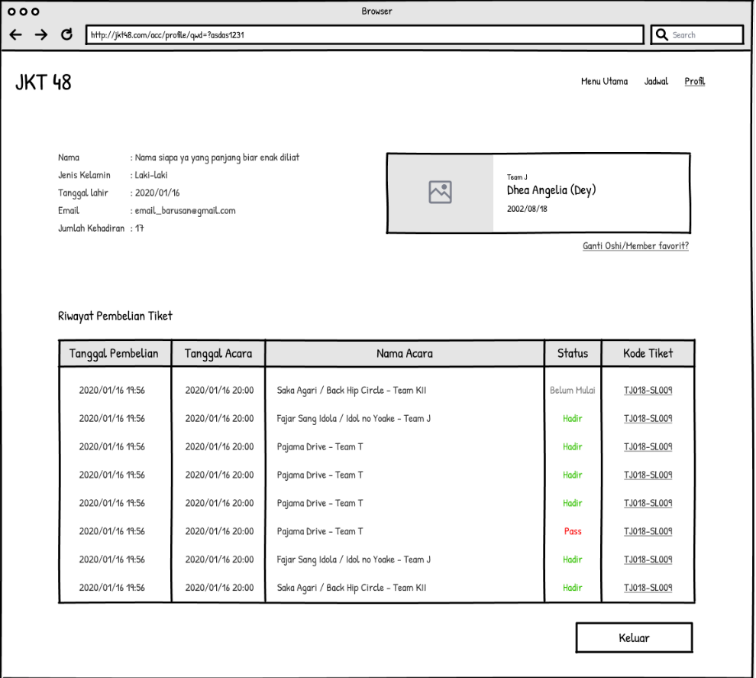
**Gambar 4.12 Menu Login (User/Fans)**

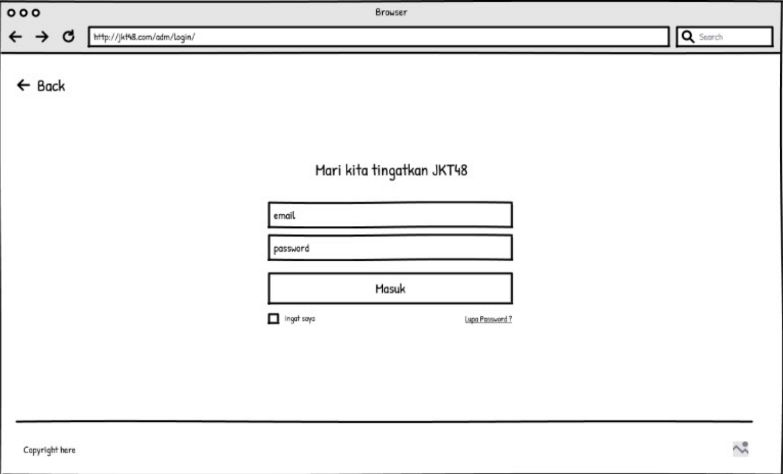
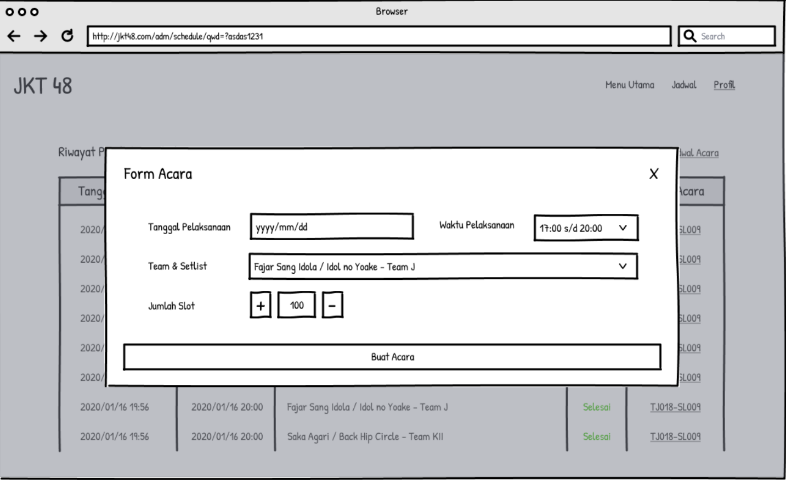
**Gambar 4.13 Menu Register Tahap 1**

****

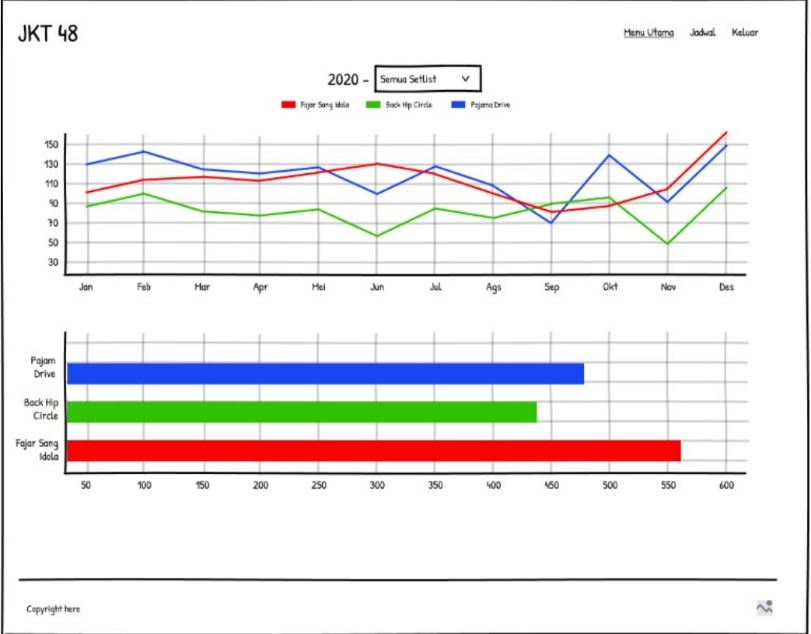
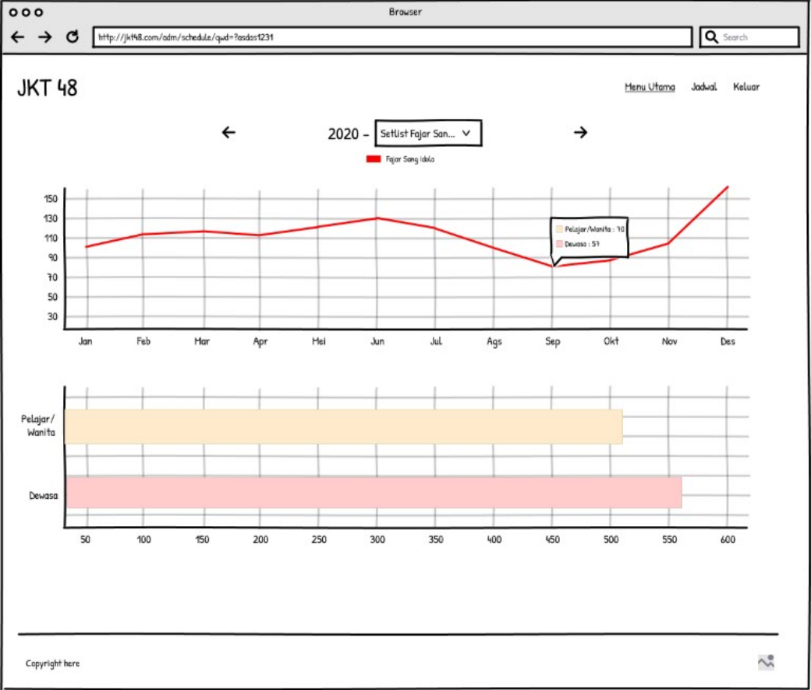
**Gambar 4.14 Menu Register Tahap 2**

**Gambar 4.15 Menu Register Tahap 3**

**Gambar 4.16 Menu Profile**

****

**Gambar 4.17 Menu Login (Admin/Staff)**

**Gambar 4.18 Menu Graphic Perkembangan Penjualan Tiket**

**Daftar Pustaka**

[1] W. N. Hidayati, “Pengalaman Komunikasi Fans JKT48 dalam Fandom JKT48,” S.Kom Skripsi, Ilmu Sosial dan Ilmu Politik, Universitas Diponegoro, Semarang, 2015.

[2] G. Y. Swara, M. Kom, and Y. Pebriadi, “Jurnal TEKNOIF ISSN : 2338-2724 REKAYASA PERANGKAT LUNAK PEMESANAN TIKET BIOSKOP Jurnal TEKNOIF ISSN : 2338-2724,” vol. 4, no. 2, pp. 27–39, 2016.

[3] I. WARMAN and R. RAMDANIANSYAH, “ANALISIS PERBANDINGAN KINERJA QUERY DATABASE MANAGEMENT SYSTEM (DBMS) ANTARA MySQL 5.7.16 DAN MARIADB 10.1,” *J. Teknoif*, vol. 6, no. 1, pp. 32–41, 2018, doi: 10.21063/jtif.2018.v6.1.32-41.

[4] U. S. Raya, “ANALISA DAN PERANCANGAN SISTEM FRONT OFFICE HOTEL,” vol. 1, no. September, pp. 1–5, 2014.

[5] E. W. Fridayanthie, T. Mahdiati, “RANCANG BANGUN SISTEM INFORMASI PERMINTAAN ATK BERBASIS INTRANET (STUDI KASUS: KEJAKSAAN NEGERI RANGKASBITUNG,” vol. 4, no.2 Desember, pp. 128–130, 2016.

[6] A. S. Rosa, M. Shalahuddin “REKAYASA PERANGKAT LUNAK TERSTRUKTUR DAN BERORIENTASI OBJEK,” vol. 7, no.1 Oktober, pp. 8–13, 2014.

[7] R. Harminingtyas, “ANALISIS LAYANAN WEBSITE SEBAGAI MEDIA PROMOSI, MEDIA TRANSAKSI DAN MEDIA INFORMASI DAN PENGARUHNYA TERHADAP BRAND IMAGE PERUSAHAAN PADA HOTEL CIPUTRA DI KOTA SEMARANG,” vol. 6, no. 3, pp. 37–57, 2014.

**LAMPIRAN**

**CREATE DATABASE**

CREATE DATABASE FinalProject;

**CREATE TABLE**

CREATE TABLE tb\_team

(

TeamId CHAR (5) NOT NULL PRIMARY KEY,

TeamName VARCHAR (10) NOT NULL,

CONSTRAINT Check\_TeamId CHECK (TeamId LIKE ('TM[0-9][0-9][0-9]'))

);

CREATE TABLE tb\_member

(

MemberId CHAR (5) NOT NULL PRIMARY KEY,

MemberName VARCHAR (50) NOT NULL,

MemberBirthday DATE DEFAULT GETDATE(),

MemberCreated DATE DEFAULT GETDATE(),

MemberPopularity NUMERIC (11,2) NOT NULL,

TeamId CHAR (5) NOT NULL FOREIGN KEY REFERENCES tb\_team (TeamId) ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT Check\_MemberId CHECK (MemberId LIKE ('MJ[0-9][0-9][0-9]'))

);

CREATE TABLE tb\_setlist

(

SetlistId CHAR (5) NOT NULL PRIMARY KEY,

SetlistName VARCHAR (20) NOT NULL,

TeamId CHAR (5) NOT NULL FOREIGN KEY REFERENCES tb\_team (TeamId) ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT Check\_SetlistId CHECK (SetlistId LIKE ('SL[0-9][0-9][0-9]'))

);

CREATE TABLE tb\_fans

(

FansId CHAR (5) NOT NULL PRIMARY KEY,

FansName VARCHAR (50) NOT NULL,

FansGender VARCHAR (10) NOT NULL,

FansBirthday DATE DEFAULT GETDATE(),

FansAttand NUMERIC (11,2) NOT NULL,

FansEmail VARCHAR (50) NOT NULL,

FansPassword VARCHAR (50) NOT NULL,

MemberId CHAR (5) NOT NULL FOREIGN KEY REFERENCES tb\_member (MemberId) ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT Check\_FansId CHECK (FansId LIKE ('FJ[0-9][0-9][0-9]')),

CONSTRAINT Check\_FansGen CHECK (FansGender IN ('Male', 'Female'))

);

CREATE TABLE tb\_staff(

StaffId CHAR(5) NOT NULL,

StaffName VARCHAR(50),

StaffSalary INT,

StaffPosition VARCHAR(20),

PRIMARY KEY (StaffId),

CONSTRAINT Check\_StaffId CHECK(StaffId LIKE 'SJ[0-9][0-9][0-9]')

);

CREATE TABLE tb\_schedule(

ScheduleId CHAR(5) NOT NULL,

DateHeld DATE,

TimeHeld CHAR(15),

ScheduleShow CHAR(1),

ScheduleLimit INT,

SetlistId CHAR(5) NOT NULL,

PRIMARY KEY (ScheduleId),

FOREIGN KEY (SetlistId) REFERENCES tb\_setlist (SetlistId) ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT Check\_ScheduleId CHECK(ScheduleId LIKE 'SD[0-9][0-9][0-9]')

);

CREATE TABLE tb\_header\_transaction(

TransId CHAR(5) NOT NULL,

FansId CHAR(5) NOT NULL,

TransDate DATE,

TransType CHAR(10),

PRIMARY KEY (TransId),

CONSTRAINT Check\_HeaderId CHECK(TransId LIKE 'HT[0-9][0-9][0-9]' AND (TransType LIKE 'Dewasa' OR TransType LIKE 'Pelajar'))

);

CREATE TABLE tb\_detail\_transaction(

TransId CHAR(5) NOT NULL,

ScheduleId CHAR(5) NOT NULL,

FOREIGN KEY (TransId) REFERENCES tb\_header\_transaction (TransId) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY (ScheduleId) REFERENCES tb\_schedule (ScheduleId) ON UPDATE CASCADE ON DELETE CASCADE,

);

CREATE TABLE tb\_ticket(

TicketId CHAR(5) NOT NULL,

TicketType CHAR(10),

TicketNo INT,

TransId CHAR(5) NOT NULL,

FOREIGN KEY (TransId) REFERENCES tb\_header\_transaction (TransId) ON UPDATE CASCADE ON DELETE CASCADE,

);

**INSERT INTO TABLE**

INSERT INTO tb\_team (TeamId, TeamName)

VALUES

('TM001', 'Team J'),

('TM002', 'Team K3'),

('TM003', 'Team T'),

('TM004', 'Academy A');

INSERT INTO tb\_member (MemberId, MemberName, MemberBirthday, MemberCreated, MemberPopularity, TeamId)

VALUES

('MJ001', 'Viona Fadrin', '2000/11/08', '2019/12/09', '0', 'TM001'),

('MJ002', 'Zhara Nur Khaulah', '2003/8/8', '2019/09/12', '0', 'TM001'),

('MJ003', 'Nabila Yussi Fitriana', '2000/12/23', '2018/10/10', '0', 'TM001'),

('MJ004', 'Riska Amelia Putri', '2000/03/18', '2019/05/10', '0', 'TM002'),

('MJ005', 'Helisma Mauludzinia Putri Kurnia', '2000/06/12', '2018/03/10', '0', 'TM002'),

('MJ006', 'Rinanda Syahputri', '2003/09/13', '2019/8/8', '0', 'TM002'),

('MJ007', 'Anglina Christy', '2005/12/5' , '2019/5/5', '0', 'TM003'),

('MJ008', 'Anastasya Narwastu Tety Handuran', '2001/06/16', '2019/07/12', '0', 'TM003'),

('MJ009', 'Sania Julia Montolalu', '2001/07/18', '2019/04/23', '0', 'TM004'),

('MJ010', 'Fransisca Saraswati Puspa Dewi', '2000/02/24', '2019/08/19', '0', 'TM004');

INSERT INTO tb\_setlist (SetlistId, SetlistName, TeamId)

VALUES

('SL001', 'High Tension', 'TM004'),

('SL002', 'Saka Agari', 'TM001'),

('SL003', 'Rapsodi', 'TM002'),

('SL004', 'Idol no Yoake', 'TM003');

INSERT INTO tb\_fans (FansId, FansName, FansGender, FansBirthday, FansAttand, FansEmail, FansPassword, MemberId)

VALUES

('FJ001', 'Faiz Azmy', 'Male', '2001/05/02', '0', 'Faizgam1ng@gmail.com', 'iloveuveva', 'MJ001'),

('FJ002', 'Veva Tudung', 'Female', '2001/03/10', '0', 'Vevatudung06@gmail.com', 'loveutoopais', 'MJ001'),

('FJ003', 'Ucok Baba', 'Male', '2003/06/03', '0', 'Ucokzz200@gmail.com', 'babazucok16', 'MJ002'),

('FJ004', 'Bambang Sugandi', 'Male', '1998/12/12', '0', 'Bambangtamvan98@gmail.com', 'tamvanbingit', 'MJ003'),

('FJ005', 'Emalia Dewi', 'Female', '2000/01/21', '0', 'Dewiliaaa3@gmail.com', 'liapakek3a', 'MJ004'),

('FJ006', 'Ryan Nixon Salim', 'Male', '2001/08/24', '0', 'Salimibu@gmail.com', 'salimbapa', 'MJ005'),

('FJ007', 'Rido Wikana', 'Male', '2002/02/14', '0', 'Wii11kan@gmail.com', 'ikankubanyak', 'MJ007'),

('FJ008', 'Lemonia Saputri', 'Female', '2004/05/18', '0', 'lemonku10@gmail.com', 'maulemonbro', 'MJ008'),

('FJ009', 'Putri Nungsep', 'Female', '2001/01/01', '0', 'putri010@gmail.com', 'parahnungsep', 'MJ010'),

('FJ010', 'Hapis Nopal', 'Male', '1999/06/06', '0', 'empisnopel@gmail.com', 'imapucekboy', 'MJ009');

INSERT INTO tb\_staff (StaffId, StaffName, StaffSalary, StaffPosition)

VALUES

('SJ001', 'Hafiz Naufal', 15000000, 'sibos'),

('SJ002', 'Putu Krisna', 12000000, 'cobos'),

('SJ003', 'Bagus Rama', 5000000, 'krocos');

INSERT INTO tb\_schedule (ScheduleId, DateHeld, TimeHeld, ScheduleShow, ScheduleLimit, SetlistId)

VALUES

('SD001', '2021/02/15', '19:00:00 PM', '1', '50', 'SL001'),

('SD002', '2021/02/16', '21:00:00 PM', '2', '50', 'SL002'),

('SD003', '2021/02/19', '13:00:00 AM', '1', '50', 'SL003'),

('SD004', '2021/02/19', '20:30:00 PM', '2', '50', 'SL004');

INSERT INTO tb\_header\_transaction (TransId, FansId, TransDate, TransType)

VALUES

('HT001', 'FJ001', '2020/12/20', 'Pelajar'),

('HT002', 'FJ002', '2020/12/20', 'Pelajar'),

('HT003', 'FJ003', '2020/12/20', 'Pelajar'),

('HT004', 'FJ004', '2020/12/6', 'Dewasa'),

('HT005', 'FJ005', '2020/12/7', 'Dewasa'),

('HT006', 'FJ006', '2020/12/20', 'Pelajar'),

('HT007', 'FJ007', '2020/12/19', 'Pelajar'),

('HT008', 'FJ008', '2020/12/19', 'Pelajar'),

('HT009', 'FJ009', '2020/12/19', 'Pelajar'),

('HT010', 'FJ010', '2020/12/4', 'Dewasa');

INSERT INTO tb\_detail\_transaction(TransId, ScheduleId)

VALUES

('HT001', 'SD001'),

('HT002', 'SD001'),

('HT003', 'SD001'),

('HT004', 'SD003'),

('HT005', 'SD004'),

('HT006', 'SD003'),

('HT007', 'SD002'),

('HT008', 'SD002'),

('HT009', 'SD002'),

('HT010', 'SD004');

INSERT INTO tb\_ticket (TicketId, TicketType, TicketNo, TransId)

VALUES

('TJ001', 'Ijo', '1', 'HT001'),

('TJ002', 'Ijo', '2', 'HT002'),

('TJ003', 'Ijo', '3', 'HT003'),

('TJ004', 'Biru', '4', 'HT004'),

('TJ005', 'Biru', '5', 'HT005'),

('TJ006', 'Ijo', '6', 'HT006'),

('TJ007', 'Ijo', '7','HT007'),

('TJ008', 'Ijo', '8', 'HT008'),

('TJ009', 'Ijo', '9', 'HT009'),

('TJ010', 'Biru', '10', 'HT010');

**VIEW**

CREATE VIEW ticket\_sales AS

SELECT \*

FROM (SELECT YEAR(TransDate) [Year],

tb\_detail\_transaction.ScheduleId,

DATENAME(MONTH, TransDate) [Month],

COUNT(1) [Sales Count]

FROM tb\_header\_transaction

INNER JOIN tb\_detail\_transaction ON tb\_detail\_transaction.TransId = tb\_header\_transaction.TransId

GROUP BY tb\_detail\_transaction.ScheduleId,

YEAR(TransDate),

DATENAME(MONTH, TransDate)) AS MontlySalesData

PIVOT( SUM([Sales Count])

FOR Month IN ([January],[February],[March],[April],[May],

[June],[July],[August],[September],[October],[November],

[December])) AS MNamePivot

CREATE VIEW view\_member\_name AS

SELECT \* FROM tb\_member ORDER BY MemberName ASC

CREATE VIEW view\_member\_birthday AS

SELECT \* FROM tb\_member ORDER BY MemberBirthday ASC

CREATE VIEW view\_member\_created\_dated AS

SELECT \* FROM tb\_member ORDER BY MemberCreated ASC

CREATE VIEW fans\_type\_validation AS

SELECT FansName [Name], FansGender [Gender], DATEDIFF(hour,FansBirthday,GETDATE())/8766 [Age],

CASE

WHEN FansGender = 'Female' THEN 'Pelajar/Wanita'

WHEN DATEDIFF(hour,FansBirthday,GETDATE())/8766 < 18 THEN 'Pelajar/Wanita'

ELSE 'Dewasa'

END [Status]

FROM tb\_fans

SELECT \*

FROM (SELECT

tb\_member.MemberName AS [Nama Member],

DATENAME(WEEK, TransDate) [Month],

COUNT(1) [Member Count]

FROM tb\_member

INNER JOIN tb\_fans ON tb\_fans.MemberId = tb\_member.MemberId

INNER JOIN tb\_header\_transaction ON tb\_header\_transaction.FansId = tb\_fans.FansId

GROUP BY tb\_member.MemberName, tb\_header\_transaction.TransDate,

DATENAME(WEEK, TransDate)) AS MemberCount

PIVOT( SUM([Member Count])

FOR WEEK IN ([Senin],[Selasa],[Rabu],[Kamis],[Jumat],

[Sabtu],[Minggu])) AS MNamePivot

**PROCEDURE**

CREATE PROCEDURE login\_JKT48 (

@FansEmail VARCHAR(50),

@FansPassword VARCHAR(50),

@FansName VARCHAR(50),

@ret INT OUTPUT

)

AS

BEGIN

SET @ret=0

SELECT @ret=1

FROM tb\_fans

WHERE (FansName=ISNULL(@FansName,NULL) OR FansEmail=ISNULL(@FansEmail,NULL)

AND [Password]=@FansPassword

END

EXEC login\_JKT48

CREATE PROCEDURE update\_fans\_presence

AS

UPDATE FansAttand SET FansAttand = FansAttand + 1 AND

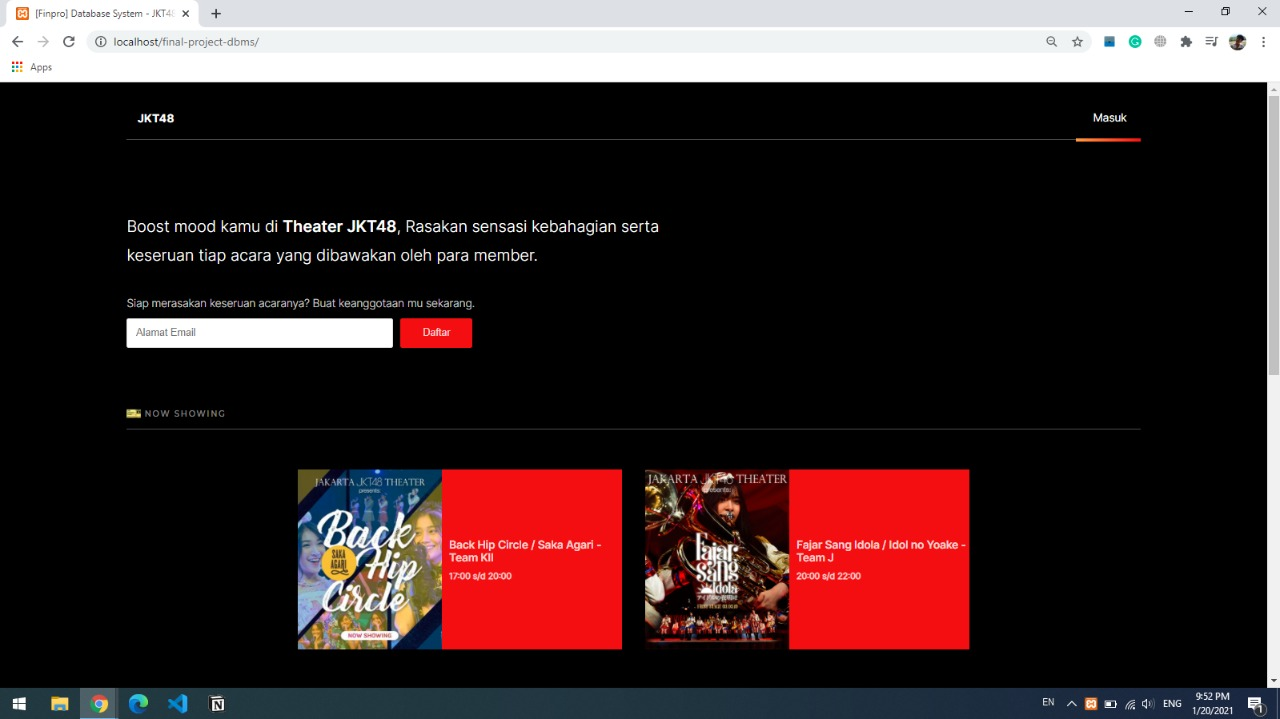
UPDATE MemberPopularity SET MemberPopularity = MemberPopularity + 1

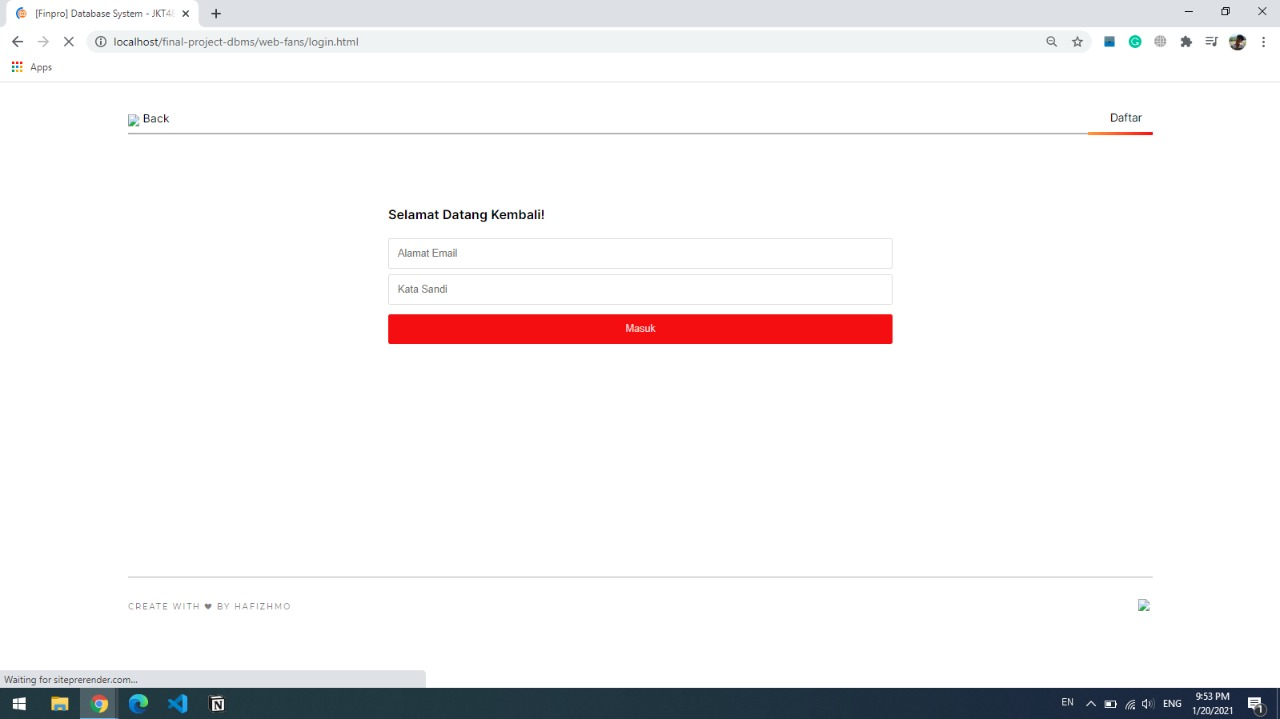
FROM tb\_fans

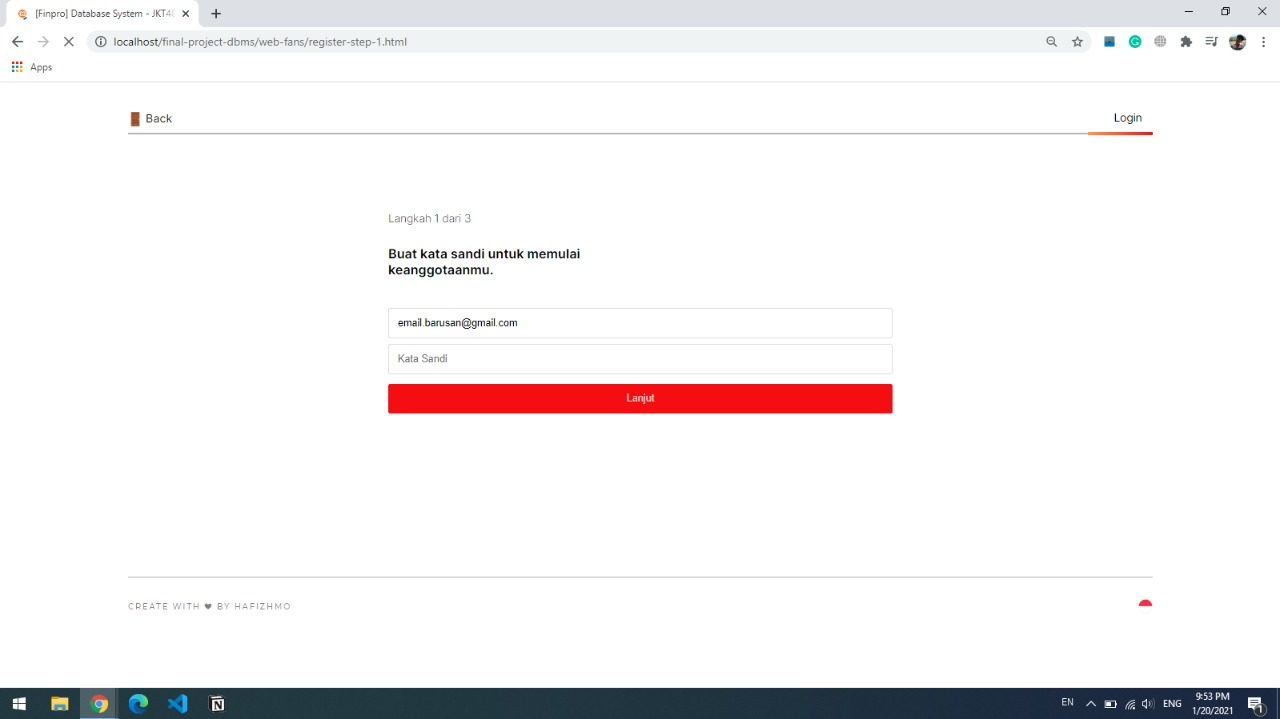
INNER JOIN tb\_member ON tb\_member.MemberId = tb\_fans.MemberId

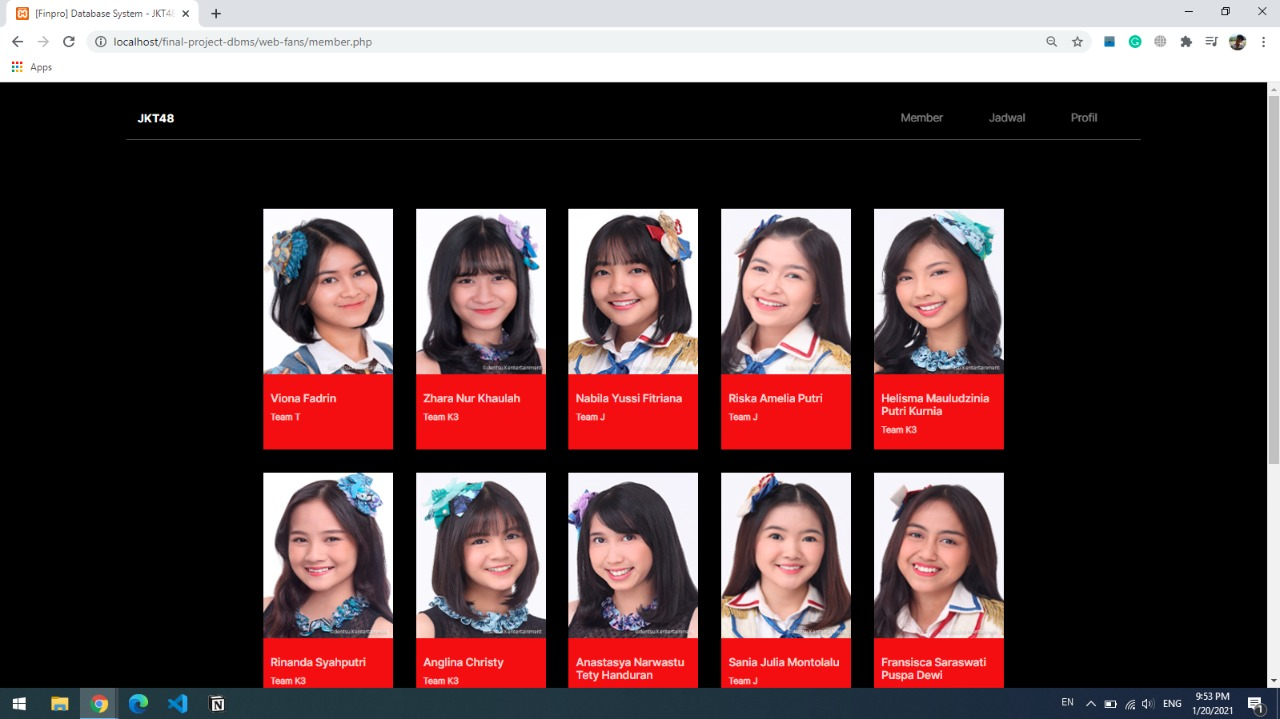
EXEC update\_fans\_presence

**WEBSITE**

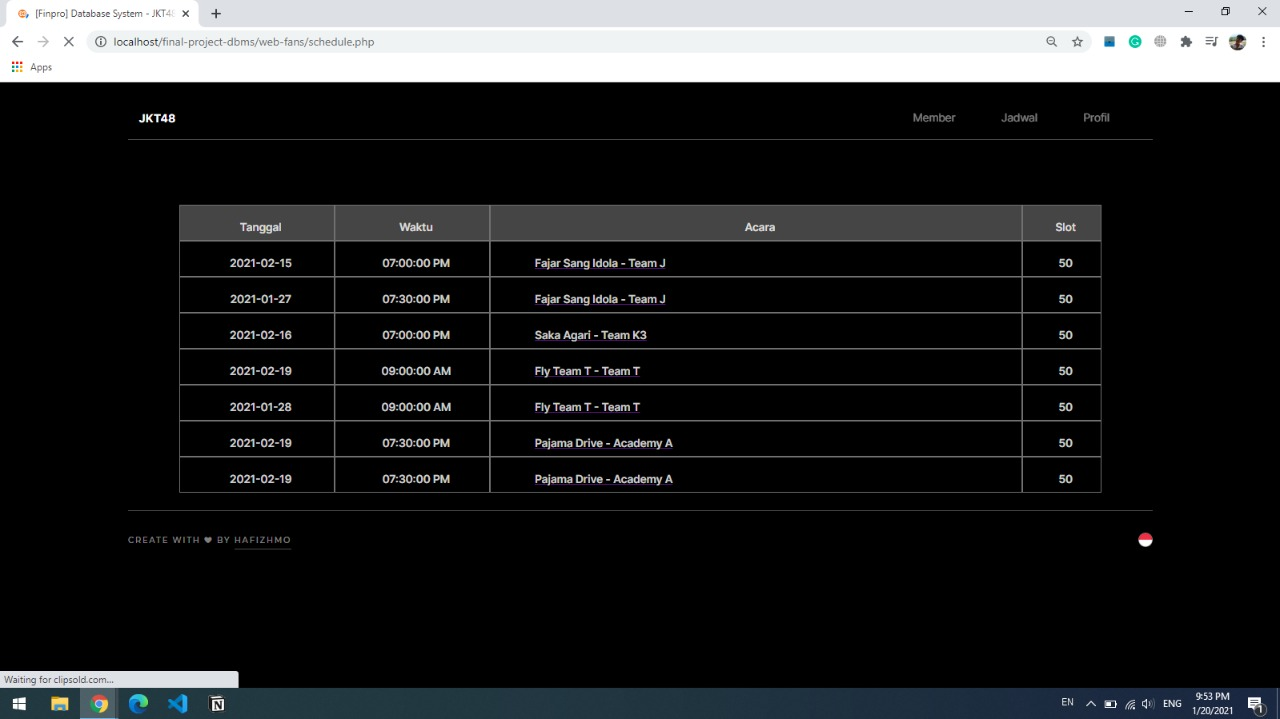
* **Halaman Home**
* **Halaman *Popular Member* dan *Group***
* **Halaman *Login***



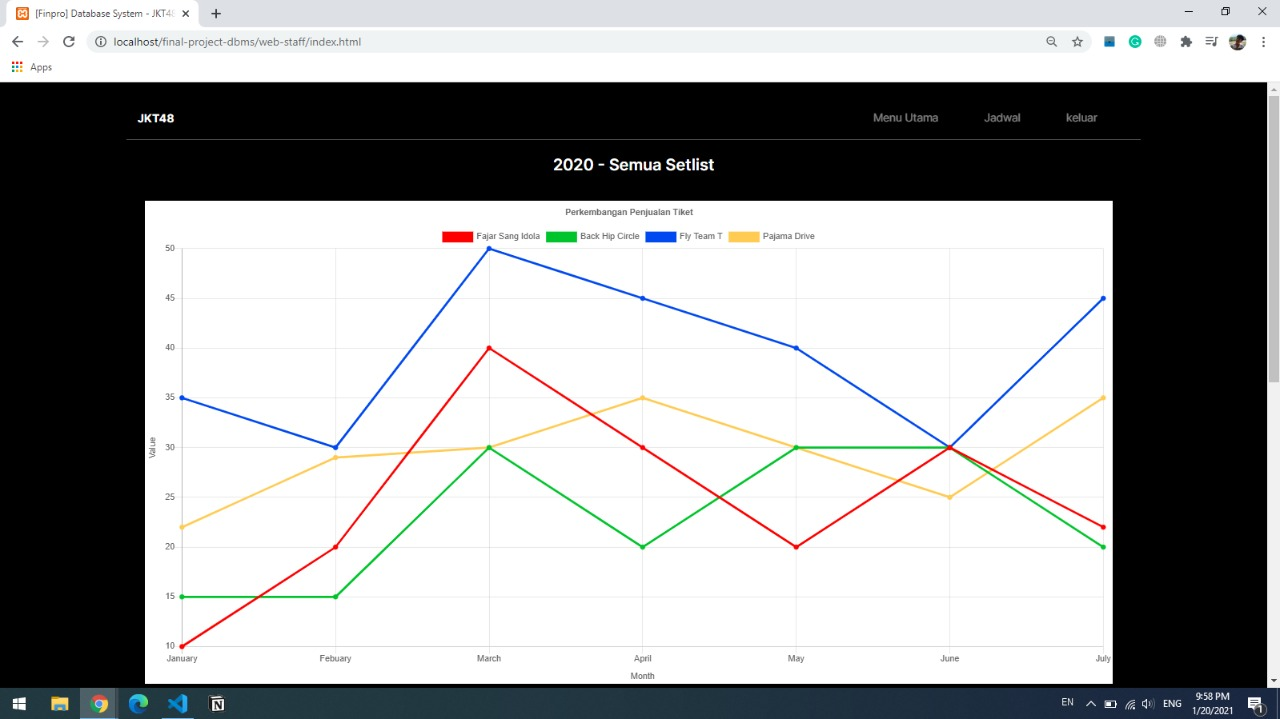
* **Halaman Pembuatan Akun**
* **Halaman Member JKT48**



* **Halaman Show Schedule JKT48**

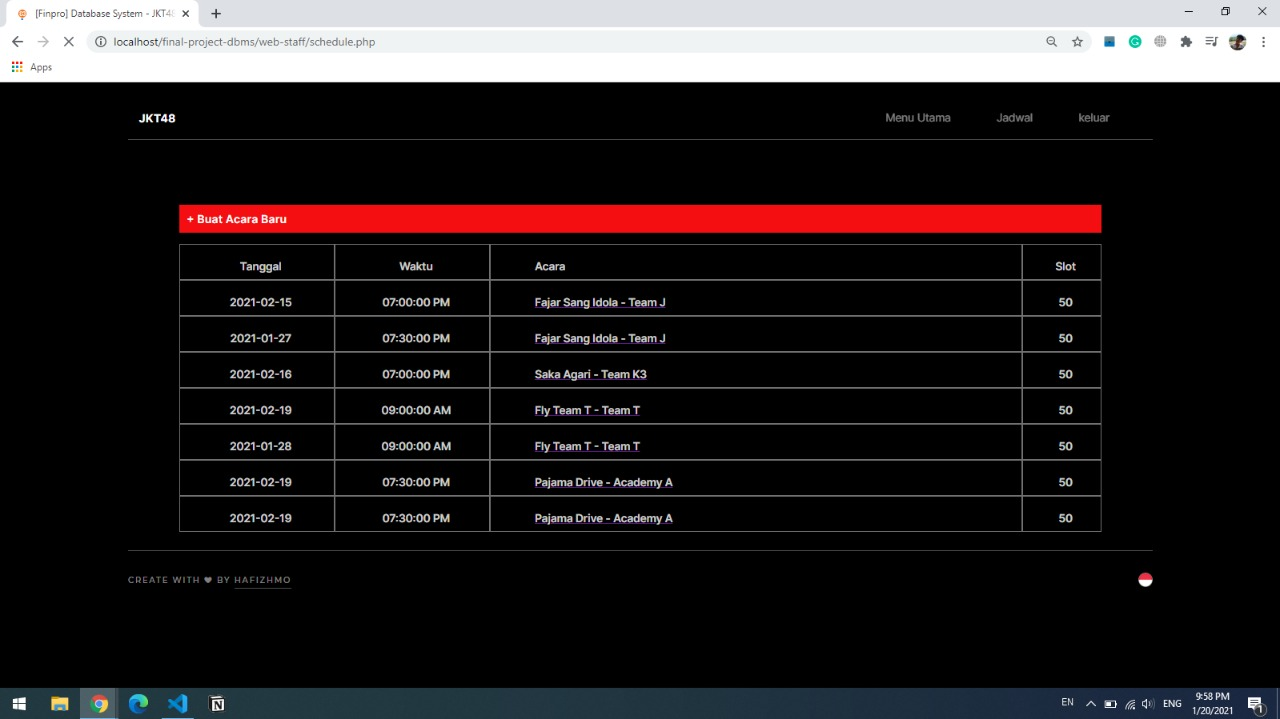


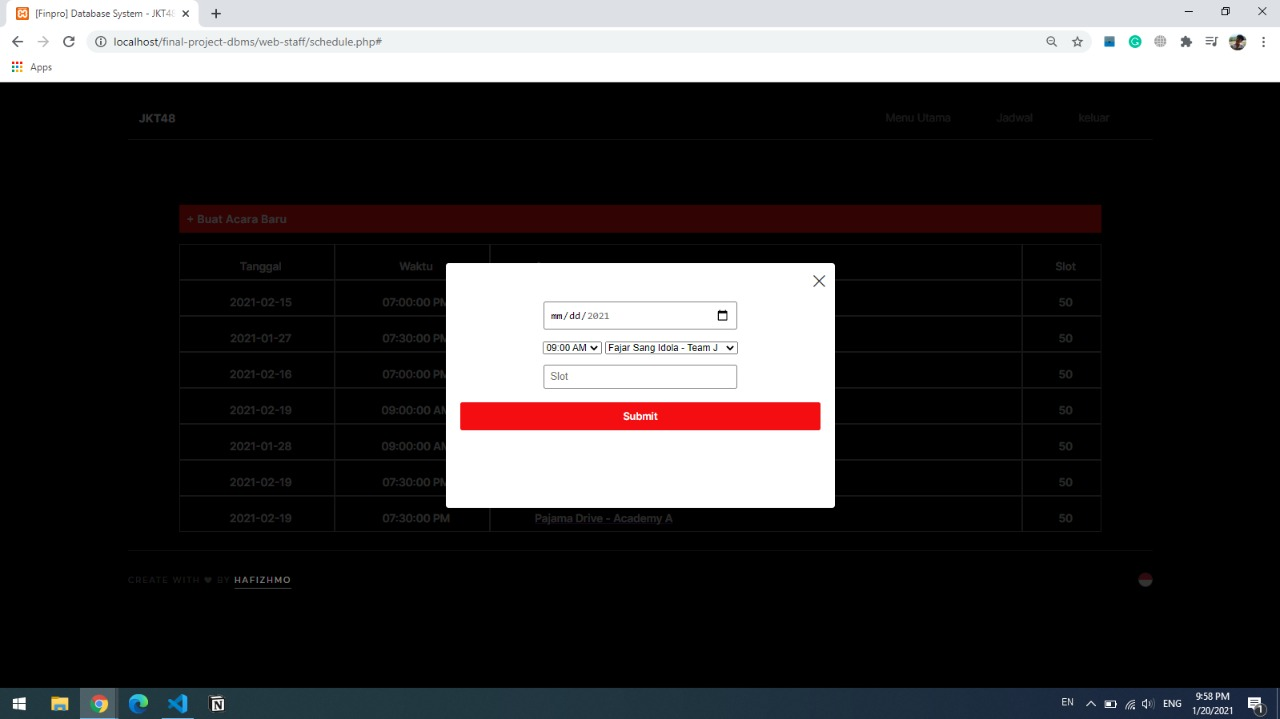
* **Halaman Persentasi Penjualan Setlist (Admin)**



* **Halaman Total Penjualan Tiket (Admin)**



* **Halaman** **Create Show (Admin)**



LEMBAR PENILAIAN

**Perancangan Database Sistem Penjualan Tiket Online Theater JKT48 Berbasis Website**

LEMBAR PENILAIAN

MATA KULIAH ISYS6028 – DATABASE SYSTEMS

KELAS BC20

Semester Ganjil 2020/2021

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DAFTAR MAHASISWA** | **NILAI** | | | | **BOBOT** | | | | **KREDIT** | | | | **TOTAL KREDIT** |
| **1** | **2** | **3** | **4** | **1** | **2** | **3** | 4 | 1 | 2 | 3 | 4 |
| I Putu Krisna Dharma Saputra - 2301924353 |  |  |  |  | 20% | 20% | 30% | 30% |  |  |  |  |  |
| Hafizh Naufal Hizbullah – 2301886310 |  |  |  |  | 20% | 20% | 30% | 30% |  |  |  |  |  |
| Ida Bagus Rama Harischandra - 2301913141 |  |  |  |  | 20% | 20% | 30% | 30% |  |  |  |  |  |
| **TOTAL** | | | | | | | | | | | | |  |

**KETERANGAN :**

* **Skala Penilaian : 0 sd 100**
* **Komponen**

1 : Laporan

2 : Produk

3 : Pengetahuan

4 : Solusi

**Malang, 20 Januari 2021**

**(Sidharta)**

**D5848**