



CHALLENGE

DOKUMENTASI TAMPILAN

CHAPTER 5

RESERVASI
TIKET BIOSKOP

TIM A CH 5



Daftar Isi

DAFTAR ISI	2
Catatan Release Dokumen	3
Version History	4
I	5
FLOW CHART AND ENTITY RELATIONSHIP DIAGRAM	5
II	7
TABLE AND INSERT DATA	11
1 II	
DOKUMENTASI TAMPILAN	11



Catatan Release Dokumen

Tanggal Release	Versi	Oleh	Catatan Perubahan
08 Juli 2022	1.0.0	Tim A CH 5	Versi Pertama

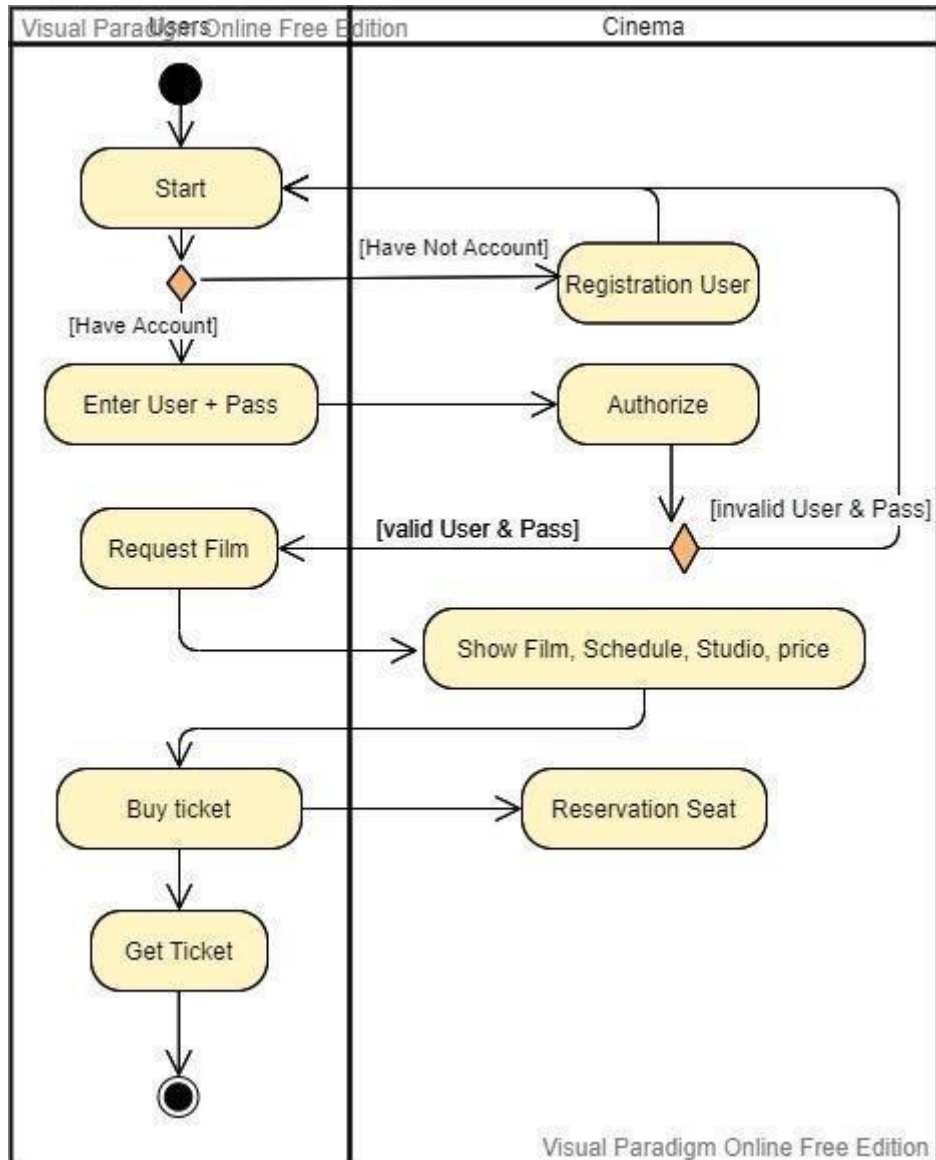


Version History

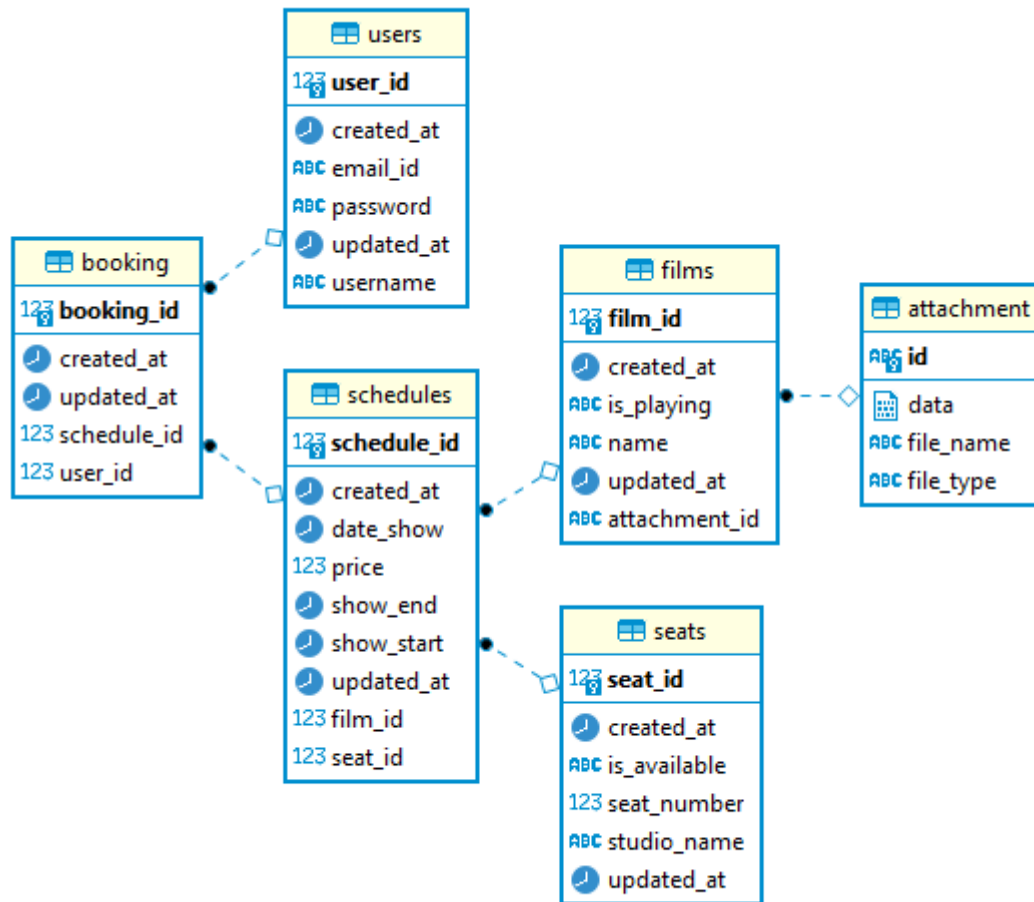
DATE	VERSION	AUTHOR	DESCRIPTION
08/07/2022	1.0.5	TIM A CH 5	DOKUMEN BERISI TAMPILAN PADA SETIAP ENDPOINT

Flow Chart And Entity Relationship Diagram

A. Project Flow



B. Entity Relationship Diagram





II

Table And Insert Data

1. Tables

- films

Column Name	Data Type	Length	Not Null	Auto Increment	Key
<i>film_id</i>	Bigserial	-	v		Primary Key
<i>Is_playing</i>	Int4	-	-		
<i>Name</i>	varchar	255	-		
<i>Created_at</i>	Timestamp	-	-		
<i>updated_at</i>	timestamp	-	-		
<i>Attachment_id</i>	Varchar	255	-		Foreign Key

Query :

```
CREATE TABLE films (
    film_id bigserial NOT NULL,
    created_at timestamp NOT NULL,
    is_playing varchar(255) NULL,
    name varchar(255) NULL,
    updated_at timestamp NULL,
    attachment_id varchar(255) NULL,
    CONSTRAINT films_pkey PRIMARY KEY (film_id)
```

);

- attachment

Column Name	Data Type	Length	Not Null	Auto Increment	Key
<i>id</i>	Varchar	255	v		Primary Key
<i>Data</i>	bytea	-	-		
<i>File_name</i>	varchar	255	-		
<i>File_type</i>	varchar	255	-		

Query :

```
CREATE TABLE attachment (
    id varchar(255) NOT NULL,
    "data" bytea NULL,
    file_name varchar(255) NULL,
    file_type varchar(255) NULL,
    CONSTRAINT attachment_pkey PRIMARY KEY (id)
```

);

- seats

Column Name	Data Type	Length	Not Null	Auto Increment	Key
<i>Seat_id</i>	Big serial	-	V		Primary Key
<i>Is_available</i>	Int4	-	-		-
<i>Seat_number</i>	Int8	-	-		-
<i>Studio_name</i>	varchar	255	-		
<i>created_at</i>	Timestamp	-	-		-



<i>updated_at</i>	datetime	-	-		-
-------------------	----------	---	---	--	---

Query :

```
CREATE TABLE public.seats (
  seat_id bigserial NOT NULL,
  is_available int4 NULL,
  seat_number int8 NULL,
  studio_name varchar(255) NULL,
  created_at timestamp(0) NULL,
  updated_at timestamp(0) NULL,
  CONSTRAINT seats_pkey PRIMARY KEY (seat_id)
);
```

- users

Column Name	Data Type	Length	Not Null	Auto Increment	Key
<i>user_id</i>	Big serial	-	v		Primary Key
<i>Email_id</i>	varchar	255	-		-
<i>Password</i>	varchar	255	-		-
<i>username</i>	time	-	-		-
<i>created_at</i>	timestamp	-	-		-
<i>updated_at</i>	timestamp	-	-		-

Query :

```
CREATE TABLE public.users (
  user_id bigserial NOT NULL,
  email_id varchar(255) NULL,
  "password" varchar(255) NULL,
  username varchar(255) NULL,
  created_at timestamp(0) NULL,
  updated_at timestamp(0) NULL,
  CONSTRAINT users_pkey PRIMARY KEY (user_id)
);
```

- schedules

Column Name	Data Type	Length	Not Null	Auto Increment	Key
<i>schedule_id</i>	Serial	-	v	-	Primary Key
<i>Date_show</i>	date	-	-	-	-
<i>price</i>	int4	-	-	-	-
<i>Show_start</i>	time	-	-	-	-
<i>Show_end</i>	time	-	-	-	-
<i>Film_id</i>	int8		-	-	foreign key
<i>Seat_id</i>	int8		-	-	foreign key
<i>created_at</i>	timestamp		-	-	-
<i>updated_at</i>	timestamp	-	-	-	-

Query :

```
CREATE TABLE public.schedules (
  schedule_id serial NOT NULL,
```




```

date_show date NULL,
price int4 NULL,
show_end time NULL,
show_start time NULL,
film_id int8 NULL,
seat_id int8 NULL,
created_at timestamp(0) NULL,
updated_at timestamp(0) NULL,
CONSTRAINT schedules_pkey PRIMARY KEY (schedule_id),
CONSTRAINT fkd4yk1lqx5o4urg84k055u89u FOREIGN KEY (film_id) REFERENCES
films(film_id),
CONSTRAINT fkq2lk4wwng0p8s5avud2f8sfg3 FOREIGN KEY (seat_id) REFERENCES
seats(seat_id)
);

```

- booking

Column Name	Data Type	Length	Not Null	Auto Increment	Key
<i>booking_id</i>	bigserial	-	v	-	Primary Key
<i>schedule_id</i>	int4	-	-	-	Foreign Key
<i>user_id</i>	int8	-	-	-	Foreign Key
<i>created_at</i>	timestamp	-	-	-	-
<i>updated_at</i>	timestamp	-	-	-	-

Query :

```

CREATE TABLE public.booking (
  booking_id bigserial NOT NULL,
  schedule_id int4 NULL,
  user_id int8 NULL,
  created_at timestamp(0) NULL,
  updated_at timestamp(0) NULL,
  CONSTRAINT booking_pkey PRIMARY KEY (booking_id),
  CONSTRAINT fk7udbel7q86k041591kj6lfmvw FOREIGN KEY (user_id) REFERENCES
  users(user_id),
  CONSTRAINT fkglyrcgklvhsgghf10kyoike3b8 FOREIGN KEY (schedule_id) REFERENCES
  schedules(schedule_id)
);

```

2. INSERT DATA

- Insert data films table

INSERT INTO films

(is_playing, "name", created_at, updated_at, attachment_id)

('value','value','value','value');

Example:

INSERT INTO films

(is_playing, "name", created_at, updated_at, attachment_id)

(1, 'Doraemon', '2022-06-01 15:00:00.000', '2022-06-01 15:00:00.000', defd994f-13fe-49ba-8d9b-7b8ec41b0805);



- Insert data attachment table

INSERT INTO attachment

(id, "data", file_name, file_type)

(value, 'value', 'value', 'value');

Example:

INSERT INTO attachment

(id, "data", file_name, file_type)

**(defd994f-13fe-49ba-8d9b-7b8ec41b0805, decode('binary-of-image','hex'), 'doraemon',
'image/png');**

- Insert data seats table

INSERT INTO seats (seat_id, is_available, seat_number, studio_name, created_at, updated_at)

VALUES ('value', 'value');

Example:

INSERT INTO seats (seat_id, is_available, seat_number, studio_name, created_at, updated_at)

VALUES

(1,1,1,'Studio A','2022-06-13 23:58:01','2022-01-07 05:08:16');

- Insert data users table

INSERT INTO users

(email_id, "password", username, created_at, updated_at)

VALUES ('value', 'value', 'value', 'value', 'value');

Example:

INSERT INTO users

(email_id, "password", username, created_at, updated_at)

**VALUES('email@email.com', 'password', 'Rudi', '2022-06-01 00:00:00.000', '2022-06-01
00:00:00.000');**

- Insert data schedules table

INSERT INTO public.schedules

(schedule_id, date_show, price, show_end, show_start, film_id, seat_id, created_at, updated_at)

VALUES ('value', 'value', 'value');

Example:

INSERT INTO public.schedules

(schedule_id, date_show, price, show_end, show_start, film_id, seat_id, created_at, updated_at)

**VALUES(1, '2022-06-01', 30000, '16:00:00', '14:00:00', 1, 1, '2022-06-01 00:00:00.000', '2022-06-01
00:00:00.000');**



- Insert data booking table

INSERT INTO booking

(created_at, schedule_id, user_id, updated_at)

VALUES ('value','value');

Example:

INSERT INTO booking

(created_at, schedule_id, user_id, updated_at)

VALUES ('2022-06-01', 1, 1, '2022-06-01'),

('2022-06-02', 2, 2, '2022-06-02')

III

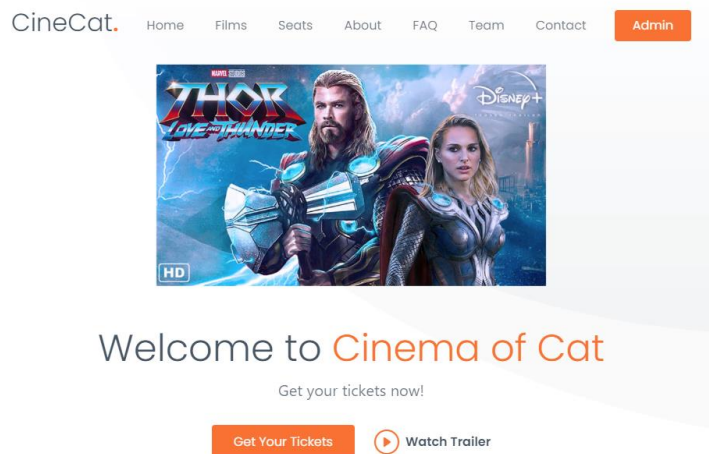
DOKUMENTASI TAMPILAN

Ringkasan :

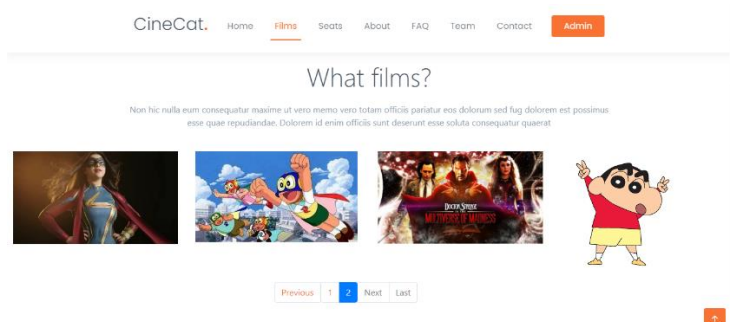
Menampilkan semua elemen tampilan yang ada pada setiap EndPoint

1. Tampilan Main Controller : Berisi Tampilan Dashboard Project

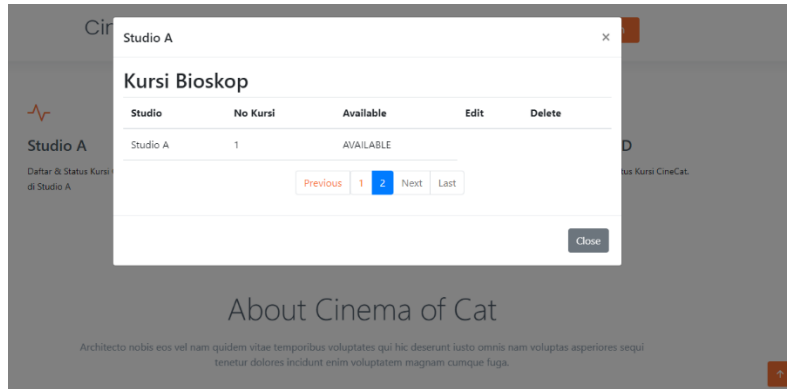
- Welcome/index/dashoard Page : Berisi Menu secara keseluruhan



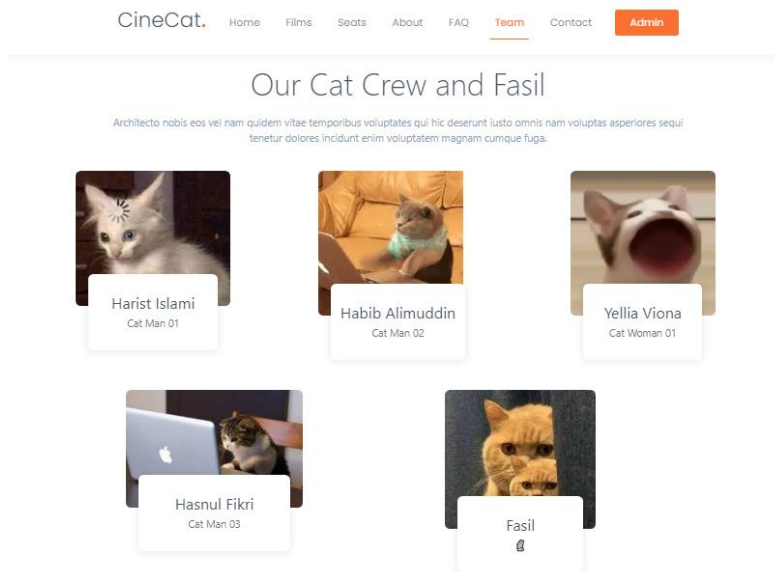
- Film List : Berisi berbagai judul film beserta informasi playing/not playing, dan sekaligus menyediakan link untuk booking ticket, serta telah memakai fitur pagination.



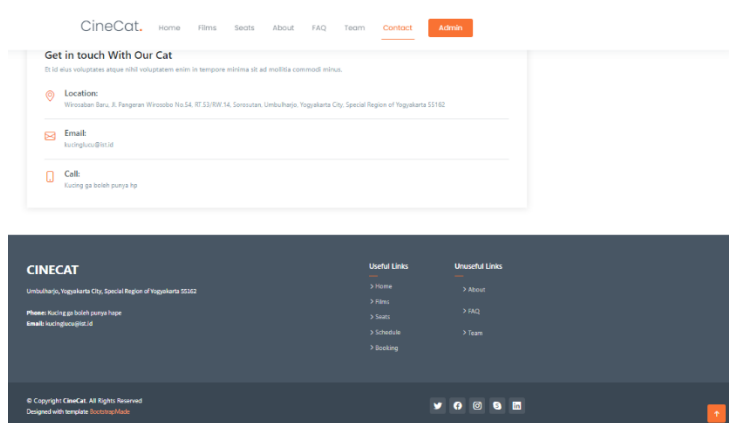
- Seats : Berisikan nama Studio, beserta daftar dan status ketersediaan kursi pada tiap-tiap Studio, serta sudah memakai fitur pagination.



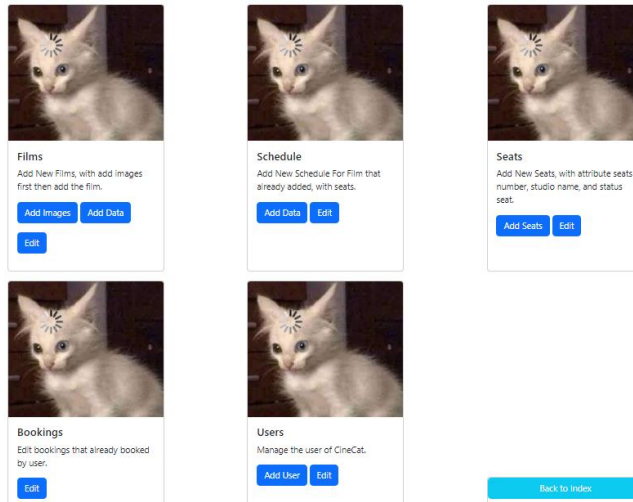
- Team : berisi semua crew dan fasil



- Contact : Berisi alamat, dan email para crew bioskop CineCat



- Halaman Admin : Berisikan semua management tentang films, schedule, seats, booking dan user. Kegunaan Halaman ini sudah dijelaskan pada setiap endpoint.



- **Film Management** : berisi tentang perintah add images, add data, & edit.
 - a) **Add images** : terlebih dahulu user diminta untuk mengupload images sesuai judul film.

Add Image For Film

No file chosen
 Choose file

- b) **Add data** : berisi tentang menambahkan judul film, status tayang, dan penginputan images untuk display di halaman dashboard.

Add New Film
















Film Name

Status


Images ▼

Didn't find the image for your film? [Add New Image Here](#)

- c) **Edit** : untuk melakukan pengeditan pada data film atau menghapus data film, sekaligus telah menerapkan konsep pagination.

Back Add files		<div>Choose</div> <div><div></div><div></div></div>		
File Name	Status	Image	Edit	Delete
Doraemon	P_L019(K)			
Naruto	NOT_PL017(K)			
One Piece	P_L019(K)			
Mortal Combat	P_L019(K)			
Chellipone	P_L019(K)			

- Schedule Management : berisi tentang penambahan data dan edit data pada schedule.
 - a. Add data : berisi tentang registrasi pada form schedule.



SCHEDULE REGISTRATION FORM

File ID Seat ID

Date Show Movie Price

Movie Start

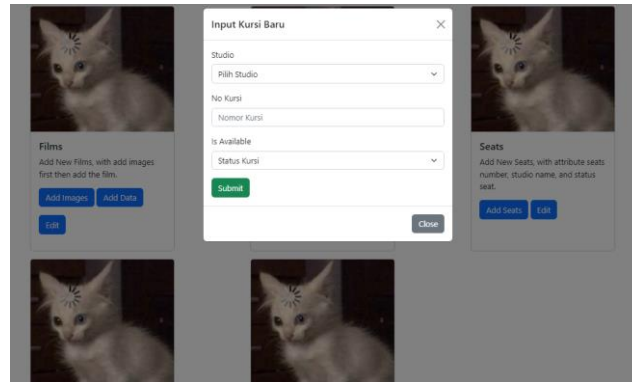
Movie End

- b. Edit : menampilkan semua movie schedule yang tersedia pada bioskop cinecat serta dapat dilakukan penghapusan schedule yg terseedia maupun edit.

Movie Schedules							

Movies	Schedules	Movie Start	Movie End	Movie Price	Seat Number	Edit Schedule	Delete Schedule
Doramen	2022-09-01	14:00	18:00	20000	1	Edit	Delete
Naruto	2022-09-02	15:00	17:00	30000	1	Edit	Delete
One Piece	2022-09-03	14:00	18:00	30000	2	Edit	Delete
Monti Combat	2022-09-04	16:00	18:00	30000	1	Edit	Delete
P MAN	2022-09-05	16:00	18:00	30000	1	Edit	Delete
Chalkzone	2022-09-06	12:00	14:00	30000	2	Edit	Delete
Inception	2022-09-07	14:00	18:00	30000	3	Edit	Delete
Terminator	2022-09-08	12:00	14:00	20000	3	Edit	Delete
Transformer	2022-09-09	12:00	14:00	30000	1	Edit	Delete
Ms. Marvel	2022-09-10	12:00	14:00	20000	2	Edit	Delete
doctor strange	2022-07-08	19:00	45:00	45000	2	Edit	Delete

- **Seat Management** : terdiri dari penambahan jumlah kursi, dan melakukan edit pada fitur seat.
 - a. **Add Seats** : melakukan input kursi baru dengan melakukan pemilihan studio, nomor kursi dan status ketersediaan kursi.



- b. Edit : opsi ini dilakukan untuk menampilkan semua kursi di masing-masing studio, sudah memakai konsep pagination, dapat dilakukan pilihan edit seat untuk melakukan update status kursi baru, dan serta dapat dilakukan penghapusan pada kursi yang tersedia pada list.

Kursi Bioskop

No Kursi	Studio	Available	Edit	Delete
1	Studio A	AVAILABLE	<button>Edit</button>	<button>Delete</button>
1	Studio B	AVAILABLE	<button>Edit</button>	<button>Delete</button>
2	Studio B	NOT_AVAILABLE	<button>Edit</button>	<button>Delete</button>
1	Studio C	AVAILABLE	<button>Edit</button>	<button>Delete</button>
1	Studio D	AVAILABLE	<button>Edit</button>	<button>Delete</button>
2	Studio D	AVAILABLE	<button>Edit</button>	<button>Delete</button>
3	Studio E	AVAILABLE	<button>Edit</button>	<button>Delete</button>
3	Studio F	NOT_AVAILABLE	<button>Edit</button>	<button>Delete</button>
1	Studio G	AVAILABLE	<button>Edit</button>	<button>Delete</button>
2	Studio H	AVAILABLE	<button>Edit</button>	<button>Delete</button>

Total Rows: 10

Previous **1** Next Last

- Bookings Management : menampilkan parauser yang sudah melakukan proses booking, beserta judul film, kursi, studio name, dan tanggal tayang, di booking managements dapat dilakukan proses edit dan delete, serta sudah mendukung fitur pagination.

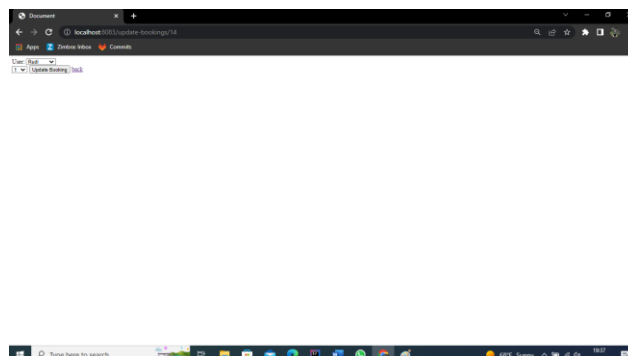
Booking

User	filname	seats	studio name	tanggaltayang	Edit	Delete
Poppy	Inception	3	Studio E	2022-06-07	<button>Edit</button>	<button>Delete</button>
Kimbepe	Chalkzone	2	Studio D	2022-06-06	<button>Edit</button>	<button>Delete</button>
Ikuro	P MAN	1	Studio D	2022-06-05	<button>Edit</button>	<button>Delete</button>
Kino	One Piece	2	Studio B	2022-06-03	<button>Edit</button>	<button>Delete</button>
Vista	Naruto	1	Studio B	2022-06-02	<button>Edit</button>	<button>Delete</button>

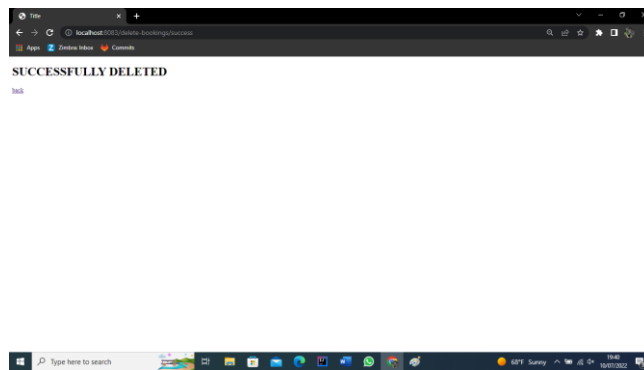
Total Rows: 14

Previous **1** 2 3 Next Last

- a. Edit : melakukan penggantian nama user, schedule id, untuk melakukan peng-update-an pada sistem booking.

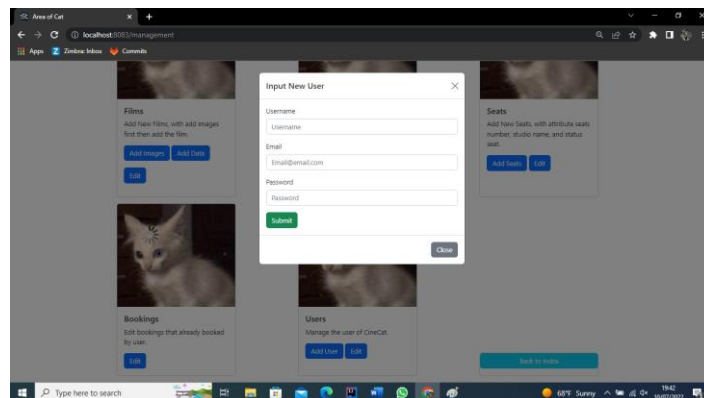


- b. Delete : melakukan penghapusan data pada sistem booking sebelumnya.



- Users Management : merupakan fitur yang disediakan untuk mendaftarkan user baru, dan edit user.

- a. Add user : daftarkan user baru.



- b. Edit : menampilkan semua data users, termasuk username, password, serta email dari masing-masing users, apat dilakukan edit dan delete pada data yang tersedia.

