

**LAPORAN UJIAN TENGAH SEMESTER
BACK END PROGRAMMING**



DISUSUN OLEH:
535220166 - Christabella Jocelynne Chandra
535220170 - Angela Chow
535220207 - Kelly Christella Gywen

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS TARUMANAGARA
2024**

BAB I

LATAR BELAKANG

1.1. Latar Belakang

Gardens by the Bay adalah tempat turisme yang terletak di Singapura. Tempat turisme ini relatif populer dengan rakyat setempat tetapi jarang dikenali bagi rakyat yang diluar singapura. Karena itu website *Gardens by the Bay* dibuat. Website ini berfungsi untuk memfasilitasi dan mempermudah proses pembookingan tiket dan sekaligus mempromosikan *Gardens by the Bay* serta memberi akses mudah kepada info-info tentang *Gardens by the Bay*. Tujuan dibuatnya website ini ialah agar menyebar dan mengenalkan nama *Gardens by the Bay* kepada potential customer yang belum mengenal *Gardens by the Bay*.

Dengan adanya website ini, jumlah customer yang datang ke *Gardens by the Bay* diharapkan menaik, karena dengan aksesibilitas yang mudah lewat internet dan juga efisiensi dari pemesanan tiket online dibanding harus dilakukan di tempat. Website ini akan menggunakan Node JS, MongoDB, Express JS menggunakan express-ejs-layouts dan CSS untuk bagian *front-end*.

BAB II

METODOLOGI

2.1.Penjelasan Metodologi

Proyek ini menggunakan metodologi *Incremental Model*, yaitu metodologi pengembangan perangkat lunak yang digunakan untuk membangun sistem yang besar dan kompleks. Metodologi ini menggunakan tahap-tahap berulang yakni Analisis, Desain, Implementasi dan Menguji.

Tahap Analisis adalah proses penentuan kebutuhan atau analisis kebutuhan. Pada tahap ini, *developer* akan melakukan berbagai macam *interview* bersama klien atau melakukan riset tentang tujuan pembuatan dan kegunaan masa depan *website*. Hal ini dilakukan untuk merencanakan fitur-fitur yang akan ada di *website*, tujuan akhir *website*, dan kompleksitas *website*.

Tahap Desain adalah proses dimana para tim *developer front-end* menentukan desain yang cocok untuk tujuan website. Desain harus bisa memikat *target audience website* tersebut atau *customer* tujuan dan juga mencakup semua fungsi-fungsi yang akan ditambahkan pada *website*.

Tahap Implementasi adalah tahap dimana para *developer* mulai menambahkan fitur-fitur yang telah ditentukan pada tahap analisis. Fitur-fitur tersebut ditambahkan per tahap, dan disesuaikan dengan permintaan klien. Hasil dari tahap ini adalah prototipe yang dapat digunakan.

Tahap Menguji merupakan tahap akhir dalam setiap *increment*, dimana prototipe akan diuji untuk mencari jika ada masalah dalam kode seperti *bug*. Jika ada, masalah tersebut akan diperbaiki pada *increment* berikutnya dan jika tidak maka prototipe akan berlangsung kepada *increment* berikutnya dimana tahap-tahap berulang dan prototipe akan ditambahkan fitur-fitur lagi.

Incremental Model memiliki tingkat fleksibilitas yang tinggi. Hal ini memudahkan *developer* untuk memperbarui (*Update*) websitenya dengan menambahkan fitur-fitur baru yang sesuai dengan permintaan dan kebutuhan pengguna.

Manajemen risiko pada *incremental model* tergolong baik, hal ini dikarenakan pengidentifikasi resiko dapat dilakukan dari sejak awal proses pengembangan, maka *developer* atau tim *developer* dapat cepat merespon dan mengatasi masalah secara efektif. Risiko yang terdeteksi pada tahap awal pengembangan juga menghemat waktu dan mencegah terjadinya kesalahan-kesalahan yang sama atau mirip pada tahap-tahap kedepannya. Dengan *incremental model*, *website* juga dapat

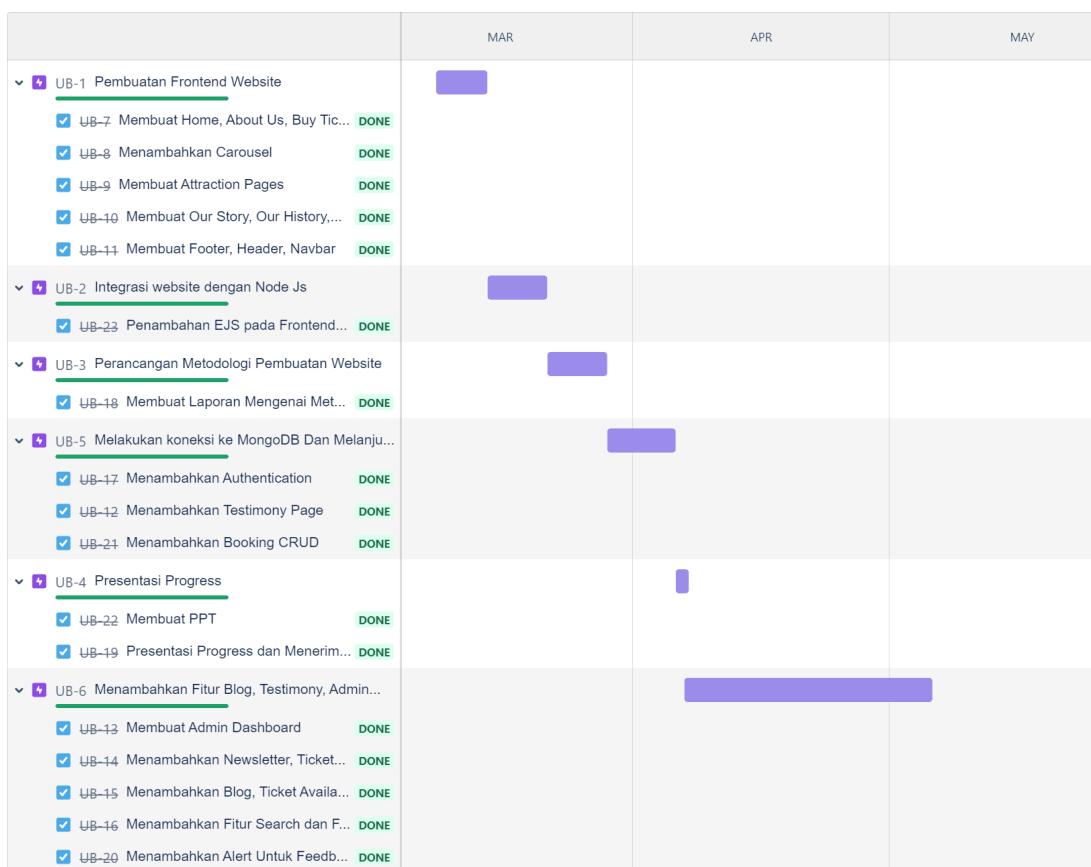
dianalisis berulang kali untuk menambahkan fitur-fitur yang kurang agar *website* lebih sesuai dengan kebutuhan pada *increment* berikutnya. Proses ini akan dilakukan berulang kali hingga *website* selesai dikembangkan pada *increment* terakhir.

2.2. Timeline Project

Timeline Project merupakan jadwal untuk seluruh projek dari mulai hingga akhir yang biasa terdiri dari beberapa tugas-tugas kecil dan satu milestone *besar*. *Timeline project* berikut yang dibuat dengan *software* Jira menggambarkan proses pengembangan *website* *Gardens by the Bay* dengan metodologi *incremental*.



Gambar berikut menggambarkan timeline project secara lebih detail.



Iterasi-iterasi selama pengembangan *website Gardens by the Bay*.

1. Iterasi pertama

Pada iterasi ini, kami mendesain dan membuat *front-end* untuk *website*. Ini termasuk semua *page-page* yang akan dibutuhkan oleh *website*, *Header*, *Footer* dan lain-lainnya.

2. Iterasi Kedua

Iterasi ini mengintegrasikan kode *front-end* yang sudah dibuat dengan Node JS dan juga mengubah kode HTML menjadi kode EJS menggunakan express-ejs-layouts.

3. Iterasi Ketiga

Iterasi ini mengambil hasil dari iterasi kedua dan melanjutkan koneksi kepada MongoDB, fitur-fitur yang ditambahkan pada iterasi ini yakni menambahkan *user authentication*, menambahkan *testimony*, dan menambahkan CRUD *booking ticket*.

4. Iterasi Keempat

Pada iterasi ini, progress dari proyek kami dipresentasikan dan dibuka untuk menerima pendapat. Sekaligus juga membuat PPT untuk menunjukkan detil-detil serta latar belakang dari proyek.

5. Iterasi Kelima

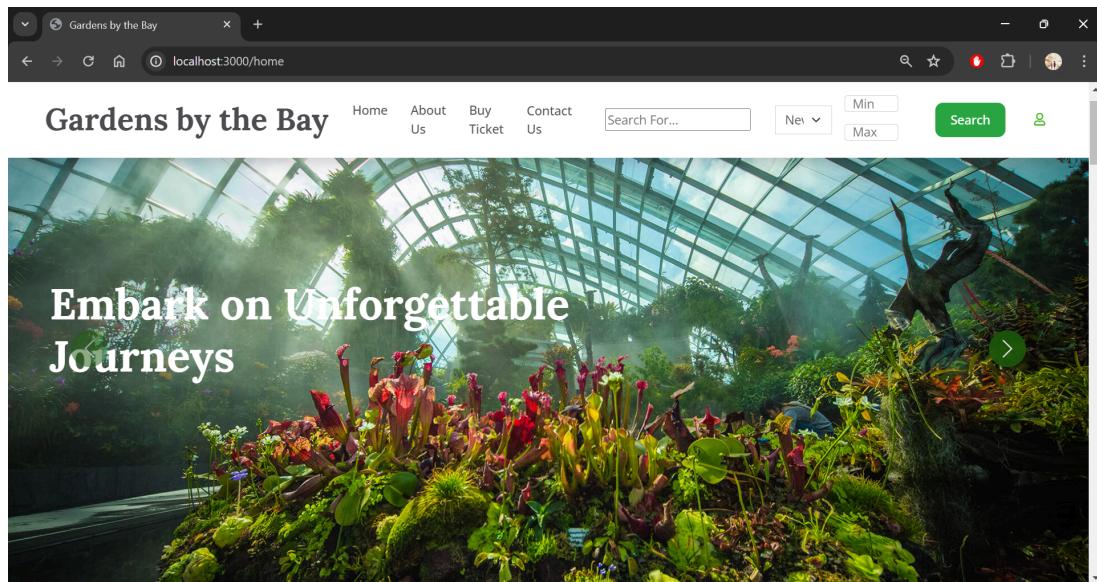
Iterasi ini sudah menunjukkan prototipe yang dapat dibilang layak digunakan. Pada tahap ini, kelompok kami sudah menambahkan *admin dashboard*, *newsletter*, *ticket availability*, *blog* dan *ticket booking* serta fitur-fitur tambahan seperti *search*, *sort*, dan *filter* dan lain lainnya.

BAB III

PROGRESS PROJECTS

1. Kondisi Terakhir Project

Website Gardens by the Bay menggunakan Node.js sebagai lingkungan *runtime* JavaScript yang memungkinkan menjalankan kode JavaScript di sisi server. *Express framework* digunakan dalam *website* ini untuk menangani permintaan HTTP, *routing*, dan lain sebagainya. Selain itu EJS (*Embedded Javascript*) menyediakan *templating* untuk mempermudah pengembangan *website* dengan Node.js dan Express.



Gambar diatas adalah *homepage* dari *Gardens by the Bay*. Pada *navigation bar* terdapat *Home*, *About Us*, *Buy Ticket* untuk memesan tiket, *Contact Us*, fitur *search*, *sort*, *filter*, dan button untuk menuju *Profile*, *Login*, dan *Signup*.

The screenshot shows the 'Gardens by the Bay' homepage. At the top, there's a navigation bar with links for Home, About Us, Buy Ticket, Contact Us, a search bar, and sorting options (Newest First, Min, Max). Below the header, there are three cards representing different attractions:

- Garden Pod**: An aerial view of a modern building integrated into a lush green landscape.
- OCBC Highway**: A vibrant aerial walkway suspended between Supertrees, with a city skyline in the background.
- Floral Fantasy**: A display of numerous colorful hanging flower baskets.

Each card includes a brief description, creation date (e.g., 5/6/2019), last update date (e.g., 5/6/2024), and a 'Read More' button.

Pada bagian tengah *homepage* terdapat list *post* atraksi yang dapat diakses untuk menuju *page post* itu tersendiri, dilengkapi fitur *pagination* untuk membatasi jumlah *post* yang muncul dalam 1 halaman dan dapat berpindah ke *post-post* yang lebih lama.

The screenshots show the 'Customer Review' section. The left screenshot displays three testimonies from users Bella, Kelly, and Kazsha, each with a quote and a short bio. The right screenshot shows a form titled 'Add Testimony' with a text input field and a 'Submit' button.

Selanjutnya pada bagian bawah *home page* terdapat bagian testimoni yang menampilkan ulasan dari *user* yang pernah mengunjungi *Gardens by the Bay*. *User* yang sudah melakukan *login* dapat menambahkan testimoni.

Gardens by the Bay

Discover the enchan... [Terms and Conditions](#)

Attraction
SuperTree Observatory

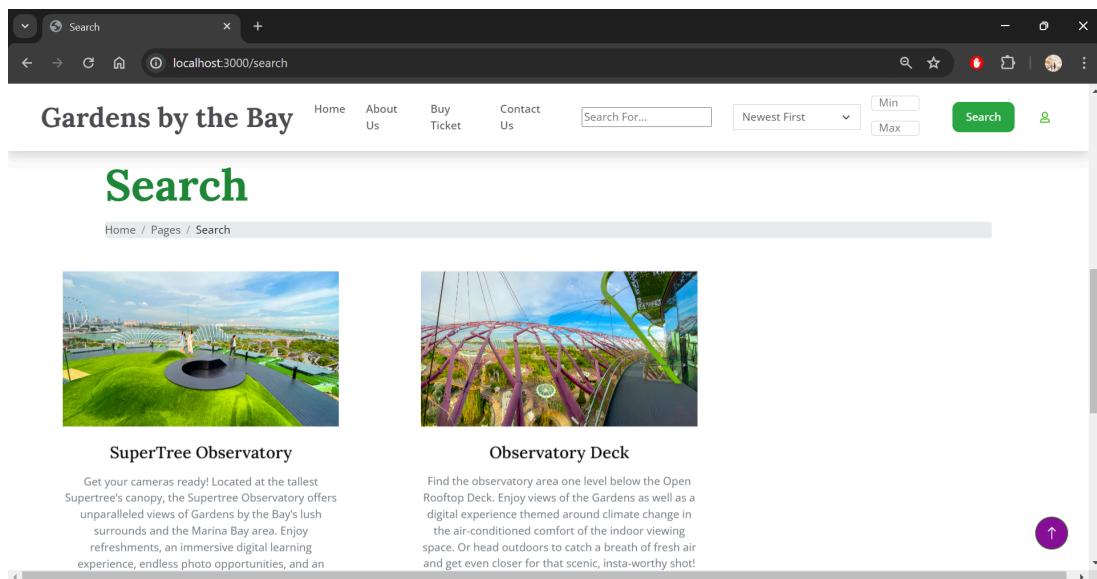
Ticket

Date
dd/mm/yyyy

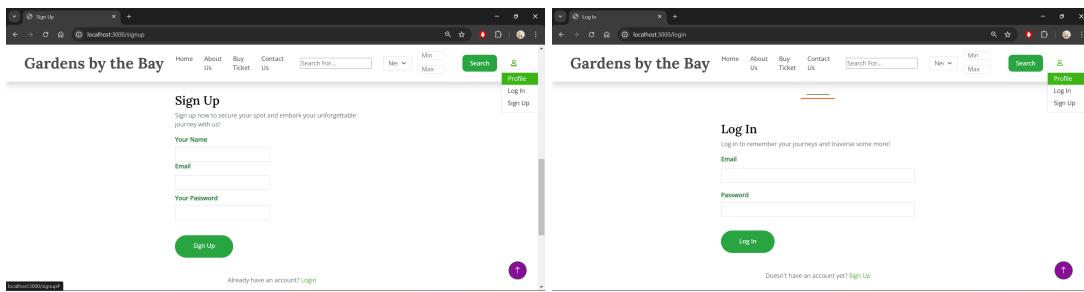
Phone number

Submit

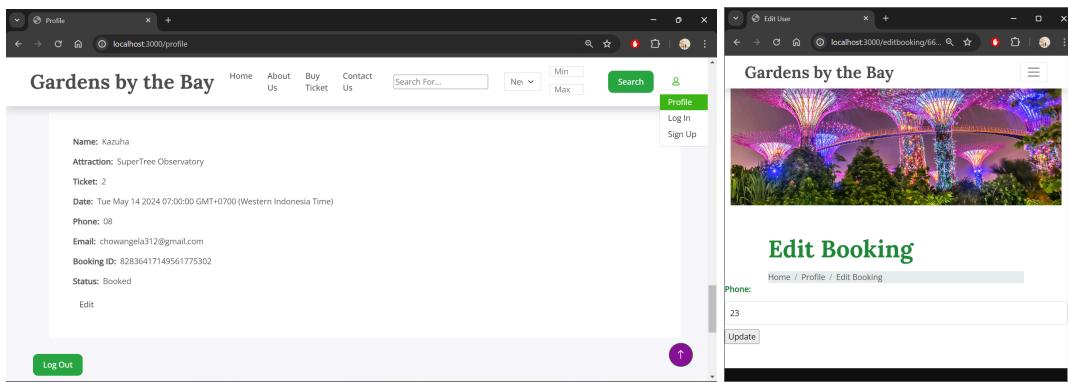
Buy Ticket Page ini hanya dapat diakses oleh *user* yang telah melakukan *login*. Dalam *page* ini *user* dapat membuat *booking* baru dengan mengisi *form* diatas.



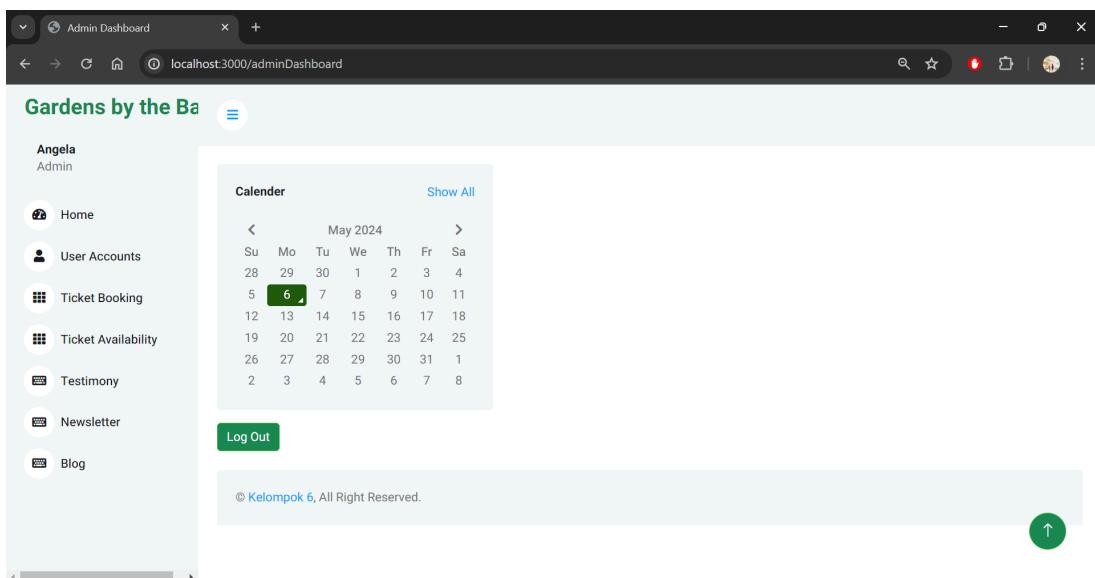
Jika *user* melakukan fitur search “*observatory*” pada *search bar* di *navigation bar*. *User* akan diarahkan ke *Search Page* yang berisi *post* atraksi dengan nama “*observatory*”. Fitur *filter* dan *sort* berguna untuk menyortir *post* berdasarkan tanggal dibuat *post* dan fitur *filter* menyaring *post* atraksi berdasarkan minimum dan maksimum harga atraksi.



Sign Up Page dan *Login Page* dapat diakses melalui *hover button* di *navigation bar*. User baru membuat akun baru dalam *Sign Up Page* dan user lama *log in* melalui *Login Page*. Halaman ini dilengkapi dengan *message feedback* kepada user setiap kali berhasil atau gagal menjalankan suatu proses.



Pada *Profile Page* diatas, ditampilkan sejarah *booking* yang pernah dibuat oleh *user* dan tombol untuk *log out* dari *website*. Detail dari *booking* juga dapat diedit dengan menekan tombol *edit*, *user* diarahkan ke *Edit Booking Page*.



Jika *user* melakukan *login* dengan akun yang berstatus admin, maka setelah *login user* akan langsung diarahkan ke Admin *Dashboard*. Admin dapat melakukan operasi CRUD dalam *User Accounts*, *Ticket Booking*, *Ticket Availability*, *Testimony*, dan *Blog*. Fitur *Newsletter* untuk mengirimkan *email* kepada seluruh *email* yang berada dalam *database*.

The screenshot shows two browser windows side-by-side. The left window is titled 'User Account' and displays a table of users with columns for Name, Email, Status, and Actions (Edit, Delete). The right window is titled 'Edit User' and shows a form for updating a user's information, specifically for 'Kazuha'. Both windows have a header 'Gardens by the Ba' and a sidebar with navigation links: Home, User Accounts, Ticket Booking, Ticket Availability, Testimony, Newsletter, and Blog.

Gambar diatas adalah *User Account Page* untuk melakukan operasi CRUD pada *user*. Pada halaman ini ditampilkan data dari *collection user*. Tombol *edit* mengalihkan admin ke *Edit User Page*, dimana nama, *email* dan status *user* dapat di *update* dan dapat melakukan *delete user*.

The screenshot shows a browser window titled 'Ticket Booking' with the URL 'localhost:3000/ticketbooking'. It displays a list of bookings on the left and a detailed view of a booking for 'Gregor' on the right. The detailed view includes fields for Name, Email, Attraction, Ticket, Date, Phone, Booking ID, and Status (Booked or Cancelled). There are 'Update' and 'Delete' buttons at the bottom. The sidebar on the left includes links for Home, User Accounts, Ticket Booking, Ticket Availability, Testimony, Newsletter, and Blog.

Gambar diatas adalah *Ticket Booking Page* untuk melakukan operasi CRUD pada *booking*. Pada halaman ini ditampilkan data dari *collection booking*. Admin dapat memperbaharui status *booking* dari *booked* ke *cancelled* dan dapat melakukan *delete booking*. *User* dapat melihat perubahan status *booking* dalam *Profile Page*.

Gambar diatas adalah *Ticket Availability Page* untuk melakukan operasi CRUD pada post. Pada halaman ini ditampilkan data *availability* dari *collection post*. Tombol edit mengalihkan admin ke *Edit Availability Page*, dimana admin dapat *update attraction*, *delete attraction*, dan *delete availability* data.

Gambar diatas adalah *Testimony Page* untuk melakukan operasi CRUD pada testimoni. Admin dapat *update* status dari *not approved* menjadi *approved* dan dapat *delete* testimoni yang tidak disetujui. Testimoni berstatus *approved* yang akan ditampilkan pada *Homepage*.

Gambar diatas adalah *Newsletter Page* beserta hasil *newsletter* yang dikirim ke *email user*. Fitur ini akan mengirimkan *email* dengan *subject* dan *message* yang telah ditentukan ke seluruh *email* dalam *collection user*.

The screenshot shows the 'Blog' section of the 'Gardens by the Bay' admin interface. The main area displays a table of blog posts with the following data:

Title	Price	Actions
SuperTree Observatory ↗	20	<button>Edit</button> <button>Delete</button>
Observatory Deck ↗	10	<button>Edit</button> <button>Delete</button>
Garden Rhapsody ↗	20	<button>Edit</button> <button>Delete</button>
Flower Dome ↗	32	<button>Edit</button> <button>Delete</button>
Cloud Forest ↗	40	<button>Edit</button> <button>Delete</button>
Floral Fantasy ↗	20	<button>Edit</button> <button>Delete</button>
OCBC Highway ↗	7	<button>Edit</button> <button>Delete</button>
Garden Pod ↗	60	<button>Edit</button> <button>Delete</button>

Below the table is a green circular button with an upward arrow. Two smaller windows are shown below the main table:

- Add New Blog**: A form with fields for Title (SuperTree Observatory) and Body (Get your cameras ready! Located at the tallest SuperTree canopy, the SuperTree Observatory offers unparalleled views of Gardens by the Bay's lush surrounds and the Marina Bay area. From here, you can also catch a glimpse of the Singapore skyline, endless phone opportunities, and an unforgettable time at the Garden's highest point).
- Edit Blog**: A form for the post 'SuperTree Observatory'. It includes fields for Title (SuperTree Observatory), Body (Get your cameras ready! Located at the tallest SuperTree canopy, the SuperTree Observatory offers unparalleled views of Gardens by the Bay's lush surrounds and the Marina Bay area. From here, you can also catch a glimpse of the Singapore skyline, endless phone opportunities, and an unforgettable time at the Garden's highest point.), ImageURL (https://www.gardensbythebay.com.sg/content/dam/gbtv/2021/vr...), TicketPrice (20), and TicketQuantity (100). A 'Save Edits' button is at the bottom.

Gambar diatas adalah *Blog Page* untuk melakukan operasi CRUD pada *post*. *Blog Page* menampilkan *data* dari *collection post*. Setiap *post* dapat di *update* datanya dengan menekan tombol *edit* yang akan mengalihkan admin ke *Edit Blog Page*, dimana data dapat diubah dalam *form* dan di *update*. Tombol *Add New Blog* pada bagian atas *Blog Page* mengalihkan admin ke *Add Blog Page*, dimana admin dapat mengisi *form* dan *create post* baru.

The screenshot shows the MongoDB Compass interface with two databases open:

- bookings**: Contains 12 documents. The right panel shows a document with the following data:


```
{
        "_id": "640f7c5940e30bf794ee754f",
        "userId": "640f7c5940e30bf794ee754f",
        "name": "SuperTree Observatory",
        "attractionName": "OCBC Highline",
        "ticketPrice": "20",
        "date": "2024-05-23T00:00:00.000Z",
        "phone": "+65 123456789",
        "seat": "A101",
        "bookingID": "1881017149287200451",
        "status": true
      }
```
- posts**: Contains 8 documents. The right panel shows a document with the following data:


```
{
        "_id": "640f7c5d7469383160e3720bc4390471",
        "title": "SuperTree Observatory",
        "body": "Get your cameras ready! Located at the tallest SuperTree's canopy, the SuperTree Observatory offers unparalleled views of Gardens by the Bay's lush surrounds and the Marina Bay area. From here, you can also catch a glimpse of the Singapore skyline, endless phone opportunities, and an unforgettable time at the Garden's highest point.",
        "ImageURL": "https://www.gardensbythebay.com.sg/content/dam/gbtv/2021/vr...",
        "TicketPrice": "20",
        "TicketQuantity": "100",
        "availableTickets": [
          {
            "id": 1,
            "status": true
          },
          {
            "id": 2,
            "status": true
          },
          {
            "id": 3,
            "status": true
          },
          {
            "id": 4,
            "status": true
          }
        ],
        "createdAt": "2024-05-05T23:19:58.477Z",
        "updatedAt": "2024-05-06T01:18:27.345Z"
      }
```

The screenshot shows two MongoDB Compass windows side-by-side. The left window displays the 'testimonies' collection with 4 documents. One document is shown in detail:

```

{
  "_id": "6409ec1c1d16337893a9b116a793f994ff",
  "userID": "6437375746d7531f51ff29",
  "name": "Angelina",
  "review": "Had an amazing time at Gardens by the Bay; the blend of art and nature is truly inspiring.",
  "status": true
}

```

The right window displays the 'users' collection with 8 documents. One document is shown in detail:

```

{
  "_id": "6409ec1c1d16337893a9b116a793f994ff",
  "name": "Angelina",
  "email": "angelinachloe12@gmail.com",
  "password": "$2b$10$zG53j3k1eGd6e6mgXEP7mku.L6oKtTufsaFaedsgLztgpopqK115",
  "admin": true
}

```

Website *Gardens by the Bay* menggunakan MongoDB untuk 4 *collection*: *bookings*, *posts*, *testimonies*, dan *users*. Operasi CRUD (*Create*, *Read*, *Update*, *Delete*) dapat dilakukan pada setiap *collection* melalui *website* yang telah dibuat.

1. *Bookings*: *Collection* ini menyimpan informasi tentang pemesanan tiket dari *user*. Data yang disimpan seperti nama pemesan, nama atraksi, jumlah tiket, tanggal *booking*, nomor telepon, *email*, *booking ID*, dan status *booking*. Status *booking* ini yang di *update* oleh admin. Setiap kali *user* membuat *booking* baru, *ticket availability* menggunakan data nama atraksi, tanggal *booking*, serta jumlah tiket yang dipesan untuk menampilkan jumlah tiket yang tersisa pada tanggal *booking* tersebut.
2. *Users*: *Collection* ini menyimpan data nama, *email*, dan *password* saat *user* baru melakukan *sign up*. Saat *user* melakukan *login*, *email* dan *password* digunakan untuk autentikasi *user* menggunakan Passport Local Strategy.
3. *Posts*: *Collection* ini berisi data tentang atraksi-attraksi dalam *Gardens by the Bay*. Judul atraksi, gambar, beserta isi teks digunakan untuk membangun *post* atraksi yang ditampilkan pada halaman *home*. Data lain seperti jumlah tiket digunakan dalam mengatur *ticket availability*. Harga tiket digunakan dalam *ticket booking*. Admin dapat meng-update isi *collection* ini untuk *edit post* atraksi, *ticket booking*, dan *ticket availability*.
4. *Testimonies*: *Collection* ini berisi testimoni yang di buat oleh *user* mengenai pengalamannya mengunjungi *Gardens by the Bay*. Admin dapat *update* testimoni mana yang akan masuk ke halaman *home* dan *delete* testimoni yang tidak disetujui.

Website Gardens by the Bay ini dilengkapi dengan berbagai macam fitur untuk meningkatkan pengalamannya penggunaan *website*, antara lain:

1. *Pagination*: Fitur bagi *user* untuk melakukan navigasi ke halaman *post* atraksi yang berbeda sesuai dengan jumlah *post* yang dibatasi.
2. *Search*: Fitur untuk mencari *post* atraksi dengan mencocokkan kata kunci dengan judul *post* atau deskripsi *post*.
3. *Sort*: Fitur untuk menyortir *post* berdasarkan tanggal dibuat *post*, dapat disortir dari *post* terbaru atau *post* terlama.
4. *Filter*: Fitur untuk filtrasi *post* berdasarkan harga tiket atraksi, dapat diatur berdasarkan minimum dan maksimum harga tiket.

5. *Newsletter*: Fitur *newsletter* ini untuk mengirimkan *email* kepada seluruh *user* yang terdapat dalam *users database* dengan menerima *input subject* dan *message email*.
6. *User Authentication*: Autentikasi *user* menjamin keamanan *user* pada saat melakukan proses *log in* dan *sign up*.

Modul-modul yang digunakan dalam pengembangan website Gardens by the Bay yaitu:

```
const express = require('express'); // express framework
const expressLayouts = require('express-ejs-layouts'); // ejs
layout templating language
const bodyParser = require('body-parser'); // parse incoming req
body for data sent in http req
const mongoose = require('mongoose'); // odm for nodejs mongodb,
schema
const path = require('path'); // for manipulating file path
cross platform
const methodOverride = require('method-override'); // override
HTTP req to support Put and DELETE req
const flash = require('express-flash'); // display flash
messages to provide feedback for user after action
const session = require('express-session'); // managing user
sessions on server side
const passport = require('passport'); // authentication
middleware
```

Modul-modul NPM diatas membantu mempercepat pengembangan website *Gardens by the Bay* dengan Node.js dan Express. Express adalah *framework* utama yang menyediakan pengaturan *routing*, penanganan HTTP, dan manajemen respons. Modul Express Layouts untuk *layout templating*, Body Parser untuk *parsing* data HTTP, Mongoose sebagai ODM MongoDB, Path untuk manipulasi jalur *file*, Method Override untuk dukungan metode HTTP, Express Flash untuk menampilkan *flash message*, Express Session untuk menangani sesi pengguna, dan Passport sebagai *middleware* autentikasi. Seluruh modul ini bekerja untuk membangun *website* yang fungsional.

2. Pembagian Tugas Anggota Kelompok

Nama Anggota	Pembagian Tugas
Christabella Jocelynne Chandra (535220166)	post CRUD; fitur <i>pagination, search, filter, dan sort; ticket booking, edit</i> video demo, PPT laporan
Angela Chow (535220170)	integrasi <i>front end</i> dengan Node.js; menerapkan EJS; fitur autentikasi dengan Passport; <i>booking</i> CRUD; <i>user</i> CRUD; <i>testimony</i> CRUD; fitur <i>newsletter; ticket availability</i> , laporan UTS
Kelly Christella Gywen (535220207)	<i>frontend</i> seluruh <i>website Gardens by the Bay</i> dari segi <i>user</i> dan admin, jira <i>timeline</i> , laporan UTS

3. Kesimpulan

Website *Gardens by the Bay* dibuat dengan menggunakan metodologi *incremental model*. Metode ini memiliki fleksibilitas tinggi yang memudahkan *developer* untuk melakukan perubahan serta *bug fix* setiap ada masalah atau ada permintaan baru dari *user*. Dengan dibuatnya *website* ini, kami sebagai anggota kelompok 6 juga belajar banyak tentang cara pembuatan *Back-End*.

Kami mengerjakan proyek ini menggunakan HTML, JavaScript, CSS, dan Bootstrap untuk membuat Front-End, dan kami menggunakan Node.js, Express, MongoDB, Passport JS, EJS serta modul-modul lainnya.

Website ini telah memenuhi tujuannya yakni untuk memfasilitasi tempat promosi *online* untuk *Gardens by the Bay* menggunakan postingan-postingan *blog* serta sebagai tempat untuk *book tiket* ke *Gardens by the Bay*.

video demo: <https://youtu.be/tOxjhmf-gI>

source code: <https://github.com/AngelaC34/Back-End>