Republic of the Philippines
BATANGAS STATE UNIVERSITY
The National Engineering University
Lipa Campus
A. Tanco Drive, Brgy. Marawoy, Lipa, Batangas, Philippines 4217
Tel Nos.: (+63 43) 980-0385; 980-0387; 980-0392 to 94 loc. 3130
E-mail Address: cics.lipa@g.batstate-u.edu.ph | Website Address: http://www.batstate-u.edu.ph

College of Informatics and Computing Sciences

In Partial Fulfillment of the Requirement IT312 - System Integration and Architecture

By:

Adaya, Angelika R.

Bautista, Edwin Angelo R.

Manalo, Jude Maverick F.

Republic of the Philippines
BATANGAS STATE UNIVERSITY
The National Engineering University
Lipa Campus
A. Tanco Drive, Brgy. Marawoy, Lipa, Batangas, Philippines 4217
Tel Nos.: (+63 43) 980-0385; 980-0387; 980-0392 to 94 loc. 3130
E-mail Address: cics.lipa@g.batstate-u.edu.ph | Website Address: http://www.batstate-u.edu.ph

College of Informatics and Computing Sciences

ACTIVITY /PROJECT
Marine Biodiversity Explorer: A Web Based Platform for Analyzing and Showcasing the Unique Characteristics of Marine Species in Tubbataha Reef

OUTCOMES AND GOAL

A. Objective(s)

The objective of Marine Biodiversity Explorer was to develop and implement a system that evaluates the ecological roles of marine species in Tubbataha Reef and analyzes user engagement to promote effective learning about its marine ecosystems as well as raising awareness for species conservation .

B. Intended Learning Outcome (ILO)

1. **Comprehensive Species Database :** Develop a detailed user-friendly database of marine species found in Tubbataha Reef , including their unique characteristics , habitats and conservation statuses with multimedia resources such as images and videos

2. **Interactive Visualization Tools:** Implement interactive tools that allow users to explore species distribution, abundance, and ecological relationships through data visualization techniques, such as maps and graphs.

3. **Community Engagement and Educational Resources:** Create a section dedicated to community involvement and education, featuring articles, videos and events that promote awareness of marine biodiversity and conservation efforts specific to the Tubbataha Reef ecosystem.

C. Sustainable Development Goal (SDG)

1. SDG 14 : Life Below Zero - this platform is about promoting marine life conserving and sustainably using oceans , seas and marine resources .

2. **SDG 4 : Quality Education** - Marine Biodiversity Explorer is aligned with quality education to promote effective learning about marine life which can also contribute to SDG 14 also known as Marine Conservation .

Republic of the Philippines
BATANGAS STATE UNIVERSITY
The National Engineering University
Lipa Campus
A. Tanco Drive, Brgy. Marawoy, Lipa, Batangas, Philippines 4217
Tel Nos.: (+63 43) 980-0385; 980-0387; 980-0392 to 94 loc. 3130
E-mail Address: cics.lipa@g.batstate-u.edu.ph | Website Address: http://www.batstate-u.edu.ph

College of Informatics and Computing Sciences

BACKGROUND / SCENARIO

Marine biodiversity refers to the variety of species and ecosystems found in the world's oceans. It plays a crucial role in maintaining the health and productivity of marine ecosystems, contributing to human well-being through food security, climate regulation, and economic opportunities. In particular, coral reefs, such as the Tubbataha Reef in the Philippines, are home to a vast array of marine species and serve as a focal point for conservation efforts.The Tubbataha Reef, a UNESCO World Heritage Site, is considered one of the most biodiverse marine ecosystems in the world. It spans over 97,000 hectares and hosts more than 600 species of fish, 360 species of corals, and numerous species of sharks, rays, dolphins, and whales. However, despite its ecological significance, there is a lack of readily accessible, organized information that highlights the unique characteristics of the marine species inhabiting this region.

The Marine Biodiversity Explorer: A Web-Based Platform for Analyzing and Showcasing the Unique Characteristics of Marine Species in Tubbataha Reef is designed to address this gap. The platform aims to serve as an educational and research tool, providing comprehensive data on the various species found in the Tubbataha Reef. By utilizing web-based technologies, the platform enables users to explore marine biodiversity interactively, offering detailed information on species classification, ecological roles, and conservation statuses.This initiative is essential in fostering public awareness and supporting research on marine conservation. By showcasing the richness of the Tubbataha Reef, the platform promotes a deeper understanding of the need to protect marine ecosystems. The tool will be valuable to educators, researchers, conservationists, and policymakers, providing a centralized resource for studying marine biodiversity and driving efforts to sustain and protect the reef’s unique ecosystem.

In comparison to other global hotspots of marine biodiversity, Tubbataha Reef has received comparatively little research attention, despite its ecological significance. While there have been many study projects, the majority of the results are published in scientific journals and technical reports, which are frequently unavailable to the general public, educators, and environmentalists. Furthermore, it might be challenging to get a thorough grasp of the biodiversity of the reef because these data sources are typically dispersed across several organizations and institutions. The efforts of educators, legislators, and conservationists who depend on centralized, easily accessible information to raise awareness, guide decision-making, and spur action for marine conservation are hampered by this fragmentation. The global threats to marine ecosystems, including overfishing, pollution, climate change, and habitat degradation, need for increased accessibility to data on marine biodiversity. Public engagement and education are crucial to generating a better knowledge of the value of marine ecosystems and boosting conservation initiatives.

College of Informatics and Computing Sciences

Digital technology advancements, especially web-based platforms, offer a great chance to close this gap by developing interactive, approachable resources that make marine biodiversity exploration and education accessible to both professionals and the general public.

The Marine Biodiversity Explorer, a web-based tool that would centralize data on the rare marine species present inside Tubbataha Reef, is envisioned as a response to this need. The purpose of this platform is to demonstrate the diversity and abundance of the reef's marine life while also acting as a resource for research and teaching. Through the use of contemporary web technologies, including interactive maps, data visualizations, and species profiles, users will be able to investigate a range of elements related to marine life in the reef, including species distribution, behavioral traits, and ecological roles, through the Marine Biodiversity Explorer.

Users will be able to explore the Tubbataha ecosystem by searching for species, analyzing species distributions, and visualizing the interactions between various marine life. The platform will also offer important details regarding ongoing research in the reef, risks to biodiversity, and conservation activities. The Marine Biodiversity Explorer seeks to involve a broader audience, including scientists, researchers, students, educators, environmentalists, and the general public, by providing this knowledge in an interactive and accessible style.

IV. CONTEXT DIAGRAM

The context diagram shows a high-level view of the system with external entities.

External Entities:

Juan Dela Cruz(User)

Process:

User Interface

Data Stores:

Login

User Registration

UserAccounts(CRUD)

Publish Articles(CRUD)

Posting & Replies (CRUD)

Dataflows:

User: Login, User Registration , UserAccounts, Publish Articles, Posting & Replies,

Logout.

DataFlow Diagram (Level 0)

1. **User Login**

   Input : Juan Dela Cruz inputs login credentials

   Process: System Verifies login credentials

   Output: Featured/Home Page (if successful) , or login error message (if login fails).

Republic of the Philippines
BATANGAS STATE UNIVERSITY
The National Engineering University
Lipa Campus
A. Tanco Drive, Brgy. Marawoy, Lipa, Batangas, Philippines 4217
Tel Nos.: (+63 43) 980-0385; 980-0387; 980-0392 to 94 loc. 3130
E-mail Address: cics.lipa@g.batstate-u.edu.ph | Website Address: http://www.batstate-u.edu.ph

College of Informatics and Computing Sciences

2. **User Registration**

input : Juan Dela Cruz inputs register credentials

Process : System confirms account

Output : Featured/Home Page (if successful) or Email address already taken/password should atleast have 18 characters. (if registration fails).

3.User Accounts (CRUD)

Input : Juan Dela Cruz views Account Page,select change password and input new password

Process : System updates user credentials

Output : Password updated successfully or password should atleast have 18 characters (if failed).

4.Publish Articles (CRUD)

Input : Juan Dela Cruz selects publish post

Process : System creates a file

Output : File Created

Input : Juan Dela Cruz edits the file

Process : System update changes to file

Input : File content updated

5. Post & Replies (CRUD)

Input : Juan Dela Cruz creates a post in discussion page

Process: System publishes post

Output : Juan Dela Cruz's post was posted on this page .

College of Informatics and Computing Sciences

Input : Juan Dela Cruz replied on his post

Process : System publishes comment

Output : Juan Dela Cruz reply was displayed on his post

External Entities 1.1:

Jude Maverick (Admin)

Process:

Admin Interface

Data Stores:

Login

Manage Users(CRUD)

Manage Articles(CRUD)

Manage Discussion(CRUD)

Manage Species(CRUD)

Data Flows:

Admin : Login , Manage Users, Manage Articles, Manage Discussion, Manage Species , Logout

Data Flow diagram (Level 0)

1. **Login**

   input: Jude Maverick inputs login credentials

   Process : System verifies login credentials

   Output : Admin dashboard (if successful) or Login Error Message (if failed successfully)

College of Informatics and Computing Sciences

2. **Manage Users (CRUD)**

   Input : Jude Maverick edits user table

   Process : System update changes to user table

   Output: User table info updated successfully

3. **Manage Articles (CRUD)**

   Input: Jude Maverick views article table

   Process: System displays database

   Output : Article database displayed

4. **Manage Discussion(CRUD)**

   Input : Jude Maverick deletes user-posts

   Process : System update changes to table

   Output : Discussion Table updated successfully

5. **Manage Species(CRUD)**

   Input: Jude Maverick adds new species to table

   Process : System update changes to table

   Output : Species table updated successfully

College of Informatics and Computing Sciences

V. SYSTEM MODEL

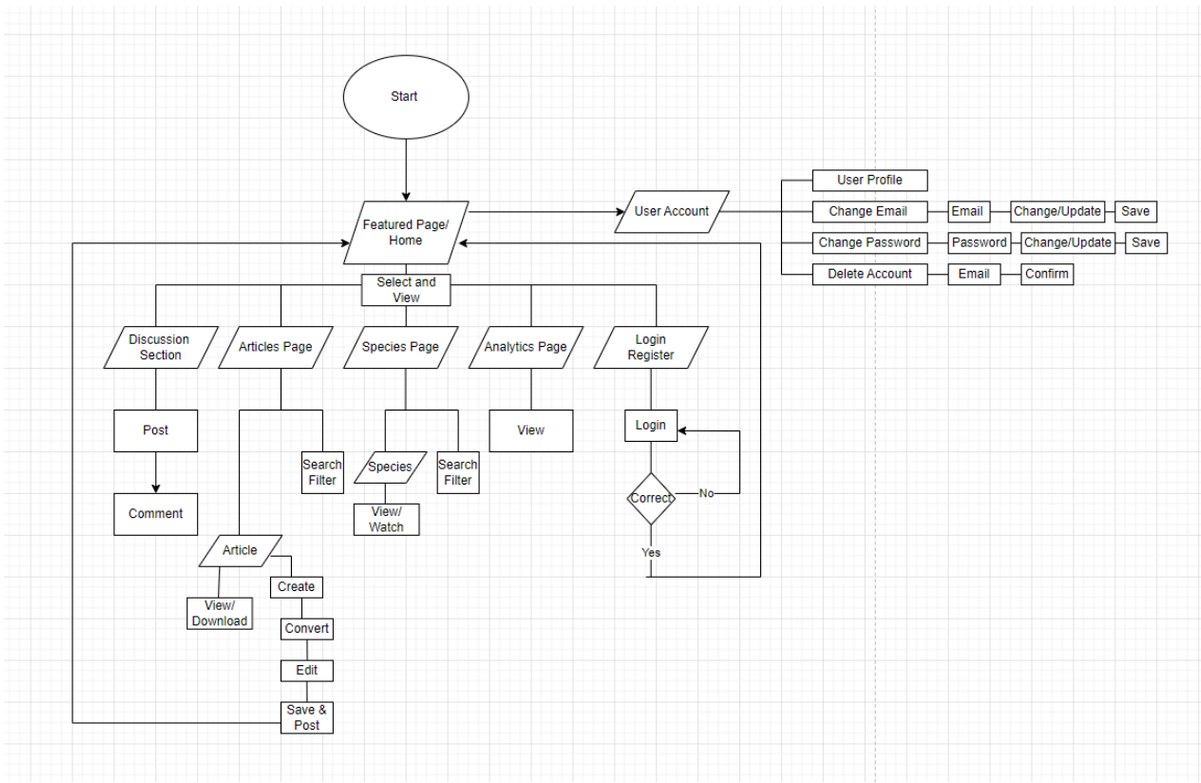


Figure 1. General User Interface

This diagram represents the General User Interface of the Marine Biodiversity Explorer web-based teaching platform, with the user being redirected to the Featured Page. When a user attempts to view the featured post on the main page or navigates to another page on the platform, they will be sent to the login/Register page. After logging in or making an account, the user will be redirected to the highlighted page, where they can choose the pages the platform has to offer. If the user has accessed the article page, they can see or download the file, as well as create, convert, and edit it. If they save changes to their file, it will be then displayed on the featured page where other users can view or download the file they created, making the platform helpful for users to understand more about marine biodiversity .

College of Informatics and Computing Sciences

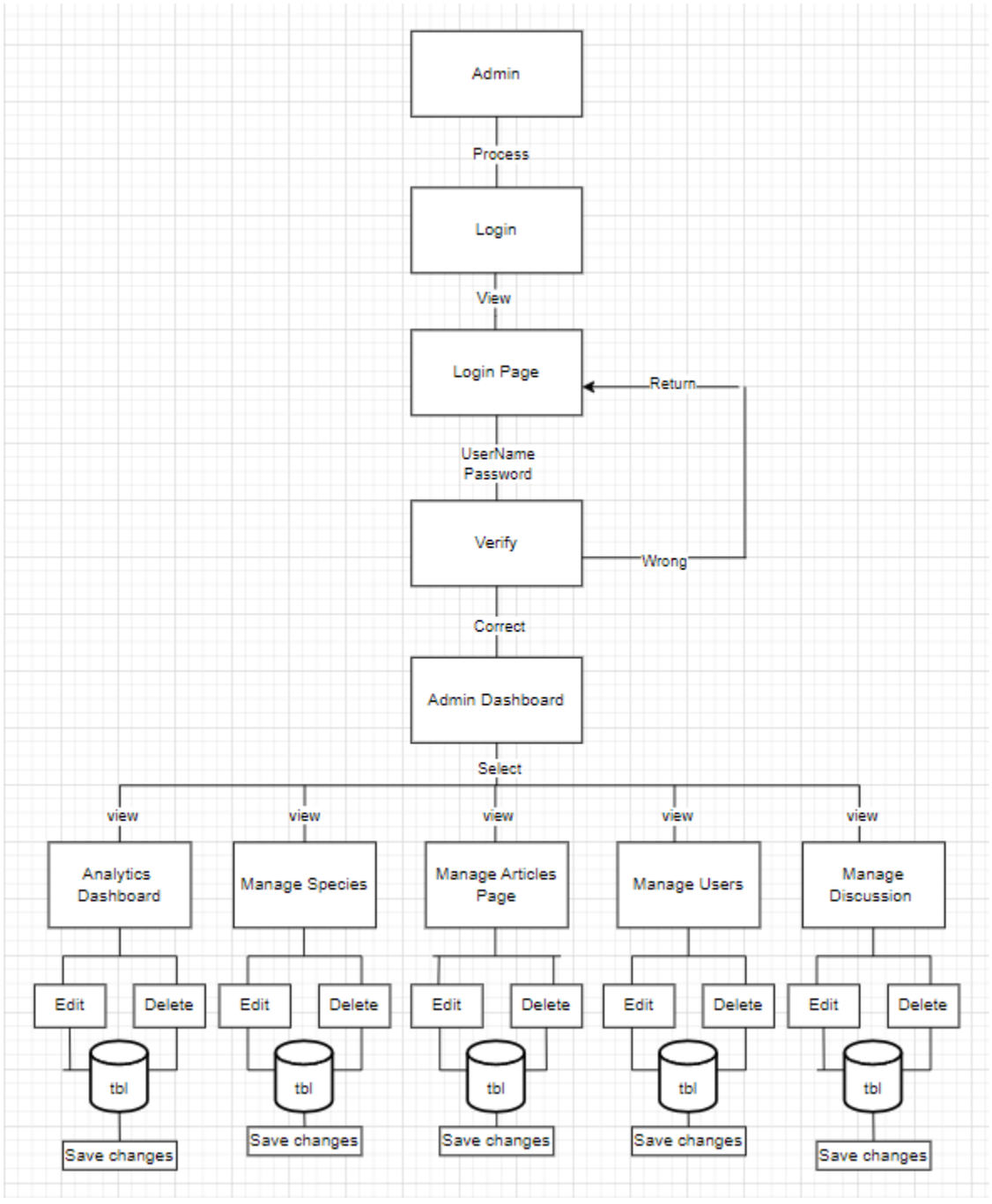


Figure 2. Admin Interface

This illustration represents how the admin interface of Marine Biodiversity explorer works. In order to access this interface , the admin must first login with correct credentials and if the password is incorrect it will return to the login page but if it is correct , it will be then redirected to the Admin Interface dashboard where it can select and view the ff: Analytics dashboard, Manage species , Manage Articles page , Manage Users , and Manage Discussion , where they can either edit or delete data.

College of Informatics and Computing Sciences

VI. DATA STRUCTURE

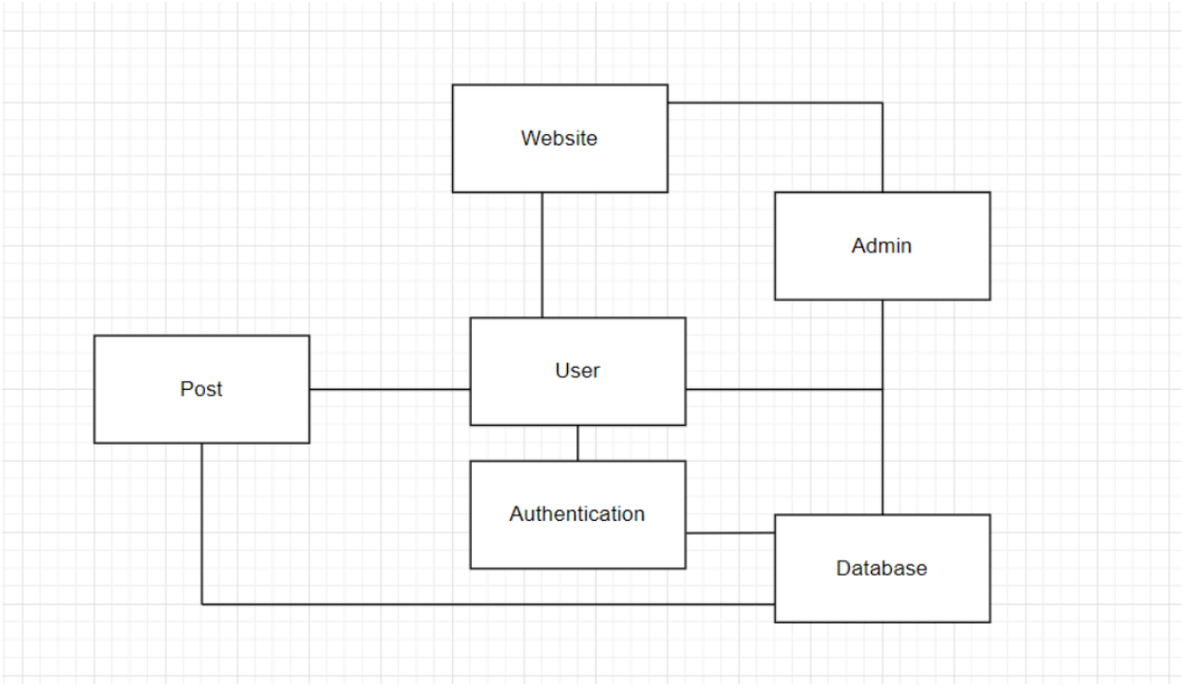


Figure 3. Web Based Educational Platform Data Structure

The diagram represents a simplified structure of a Marine Biodiversity Explorer web-based platform with various components interacting to provide website, admin, user, post, authentication and database. It outlines how the user interacts with the website, how authentication works, and how data flows between different parts of the system, like posts and the database.

Entities and Relationships:

1. Users can create , convert and edit an article via Article table.
2. Users can create a post or comment via Discussion tab
3. Admin manages the content of the platform through the database
4. Admin can remove content from the species table which can make changes to the display on the species page of the general user interface.
5. Each post and reply is associated with a timestamp for the user to know what time other users created the posts.

Species Table

College of Informatics and Computing Sciences

This table organizes the name and information of species showcased on the Species page of General User Interface of the platform .

Field Name	Data Type	Description
speciesID	INT (PRIMARY KEY)	Unique identifier for each species
speciesName	VARCHAR(100)	Common name of the Species
ScientificName	VARCHAR (150)	Scientific Name (Latin)
habitat	VARCHAR(255)	Habitat description(eg. corals)
speciesDescription	VARCHAR(max)	Detailed description of the species
Species_img_url	VARCHAR(255)	URL to an image of the species

Articles Table

This table organizes the information about the articles created by other users

Field Name	Data Type	Description
articleID	INT(PRIMARY KEY)	Unique identifier for each article
articlename	VARCHAR(100)	Name of Article
article_desc	VARCHAR (500)	Information of what the article is all about
userID	INT (FOREIGN KEY)	Foreign key referencing which user created and published the article .

Users Table

This table organizes the account credentials created by the user to access platform content

Field Name	Data Type	Description
userID	INT(PRIMARY KEY)	Unique identifier for each user
email	VARCHAR (100)	User's email address required for logging in and creating an account
FirstName	VARCHAR (100)	Name of the user which will be displayed after logging in to the platform

LastName	VARCHAR (100)	Name of the user
password	VARCHAR (20)	encrypted password for the user account

Admin Table

this table organizes the stored admin credentials to access admin interface

Field Name	Data Type	Description
adminID	INT (PRIMARY KEY)	Unique identifier for each admin
adminfname	VARCHAR (100)	Name of the user , will be displayed as username after loggin in
adminlname	VARCHAR (100)	Name of the user
admin_email	VARCHAR (80)	Admin's email address required for logging in to access admin interface
admin_passwrd	VARCHAR (20)	encrypted password for the admin account

Post Table

This table organizes posts created by users from the forum / Discussion section of the platform that includes a timestamp on what day and time the user created the post .

Field Name	Data Type	Description
postID	INT (PRIMARY KEY)	Unique identifier for each post
userID	INT (FOREIGN KEY)	Foreign key referencing which user created a post
adminID	INT (FOREIGN KEY)	Foreign Key for admin to edit or delete data from post table
created_at	TIMESTAMP	Time and date of the created post
body	VARCHAR (500)	information of the post created by the user

College of Informatics and Computing Sciences

Replies Table

This table is connected to post table since it will organize the replies of user to each post created to the forum section

Field Type	Data Type	Description
reply_id	INT(PRIMARY KEY)	unique identifier for each reply
reply_body	VARCHAR (500)	information of the replies by the user from other created posts
userID	INT (FOREIGN KEY)	Foreign key referencing which user replied on the post created by another user
adminID	INT (FOREIGN KEY)	Foreign key for admin to access and edit the data of reply table
reply_created_at	TIMESTAMP	Time and Date of the comment

Relationships:

1. **Users ↔ Articles - One user can publish multiple articles (one to many)**
2. **Users ↔ Posts - One User can create multiple posts (one to many)**
3. **Users ↔ Replies - One user can create multiple replies on every post (one to many)**
4. **Admin ↔ User - Admin can modify , save or delete the contents created by the user in the database through admin interface (one to many)**
5. **Admin ↔ Species - Admin can update and delete content in species table database(one to many)**
6. **Admin ↔ Analytics - Admin can edit ,update and delete cells from excel table for analytics (one to many)**

VII. WIREFRAME

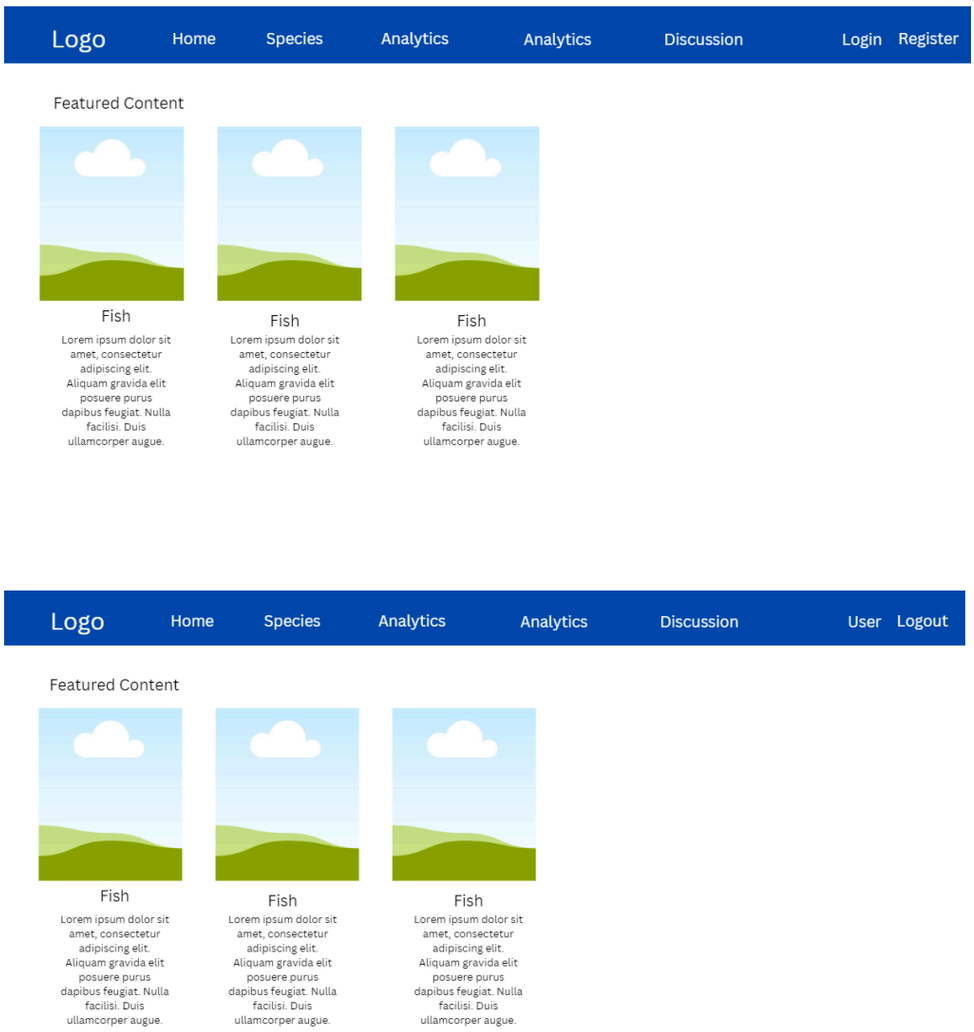


Figure 4. Featured/Home Page

If they don't create an account, the first image shows the guest account. Because they require an account to read all of the featured content on our website, they can only see the featured content. If they click on any of the highlighted content, they will be taken directly to the Login page. Additionally, the user can browse the courses within the highlighted material and view all of the featured content on the main page after logging in to their account. The featured / home page of Marine Biodiversity platform will be the landing page of the platform . Courses from this page include contents from Species page , Articles page and About us section where users can see who are the developers of the platform as well as their contact information.

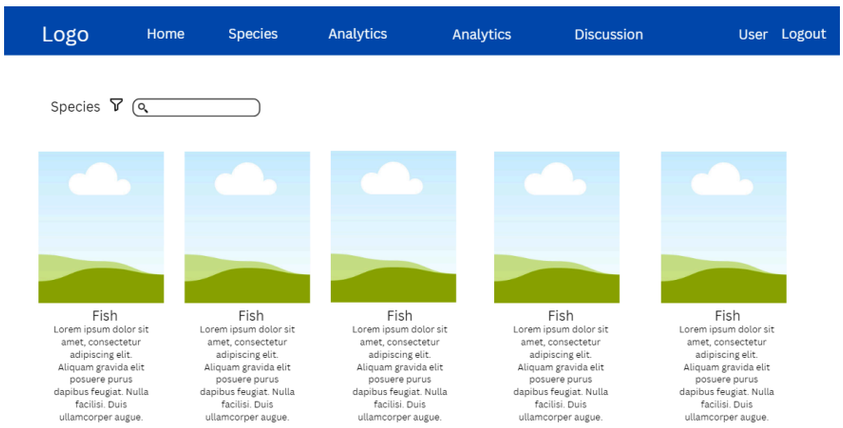


Figure 5. Lesson Page

This wireframe represents the Lessons page of Marine Biodiversity Platform where users can select which species they want to learn from. They can either search for the specific name of the species or filter it according to species class type.

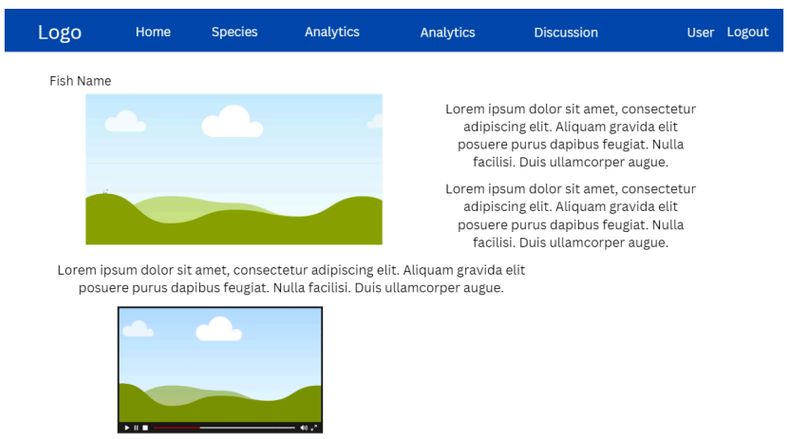


Figure 6. Species Content Page

After the user selects the content they chose in the Lesson page , they will be redirected to the content page of the specific species they selected . On that page , users can read information of the species, visual representations of what it looks like , their type of ecosystem they belong to , the species class type and some sustainable practices they can do to help conserve these species .

Figure 7. Analytics Dashboard Page

This wireframe presents the Data Analytics report of Tubbataha Reef National Park where users can view the total count of the specific species class type living in the area .

Figure 8. Article Page

This wireframe presents the article page of Marine Biodiversity Explorer platform where the user can search articles , view and download them . Users can also publish their articles through the Publish Post button or create and convert a file , edit it and save changes . The article created by users will be showcased in the landing page and article page after publishing it.

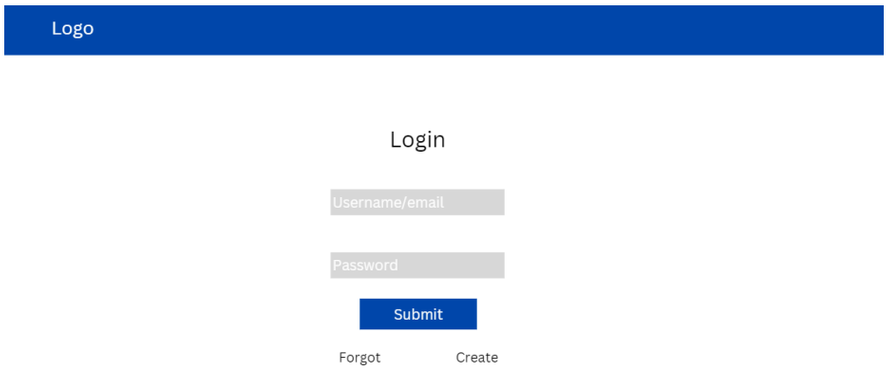


Figure 9 Login Page

This page serves as the login for users who have accounts on this website. To verify that they are the actual person entering into the account, the user will enter their username or email address and password before clicking "Submit." Click the forgot to know the forgotten password link if you can't remember your password, or make a new one. Simply click the establish button to establish an account if none already exists.

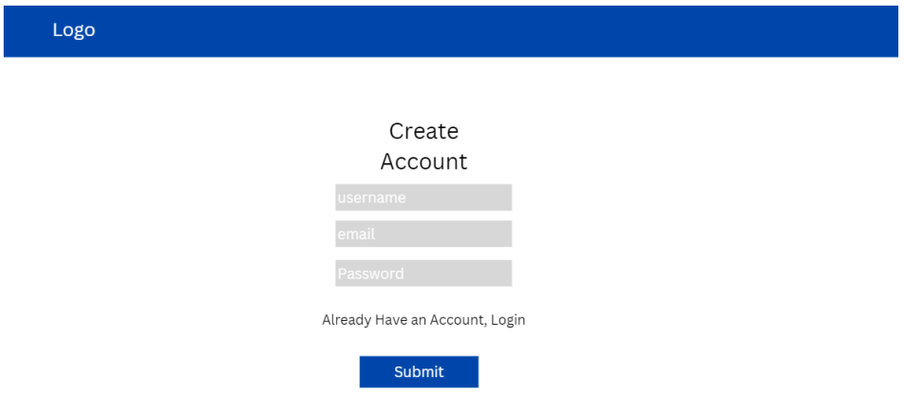


Figure 10. User Registration Page

This wireframe represents the User registration page where the user can create an account to access the contents of the platform . If the user already has an account . user may select a login and they will be redirected to the login page .

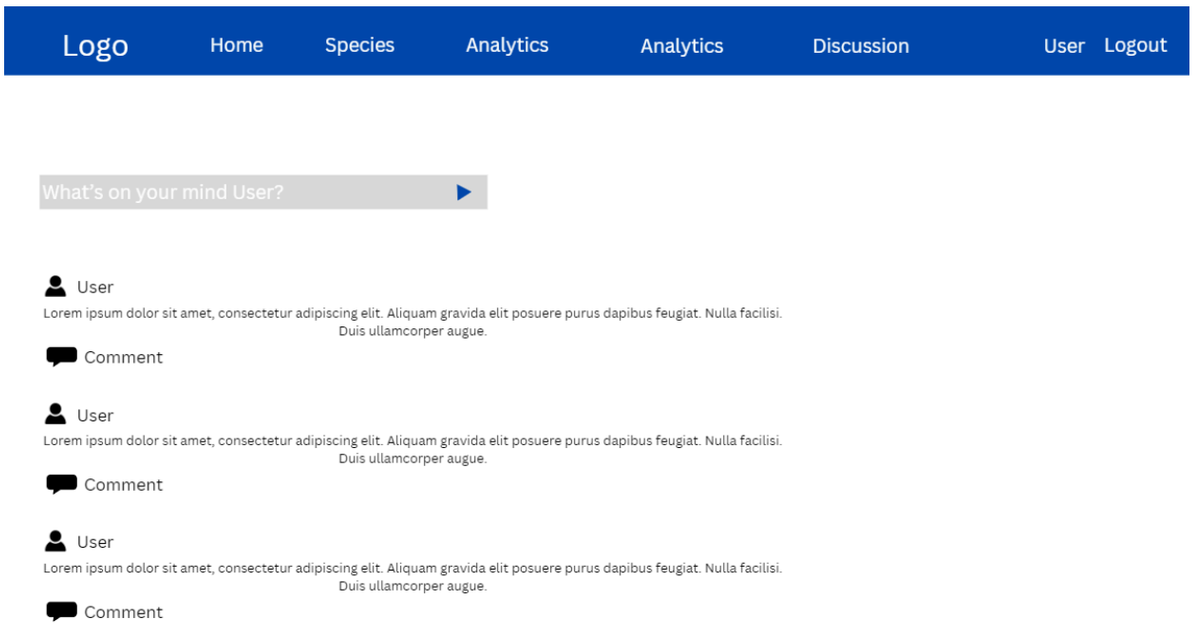


Figure 11. Discussion Page

This page is dedicated to users who are publishing discussions. Other users are welcome to post discussions of their own or to reply to those of other users, increasing the volume of discussions regarding marine biodiversity in this section.

Figure 12. Admin Interface

This wireframe represents the admin interface where the admin controls the whole website and can view every user's project via the database, and the administrator can make, remove, alter, and save modifications. The administrator can control what takes place on this website.

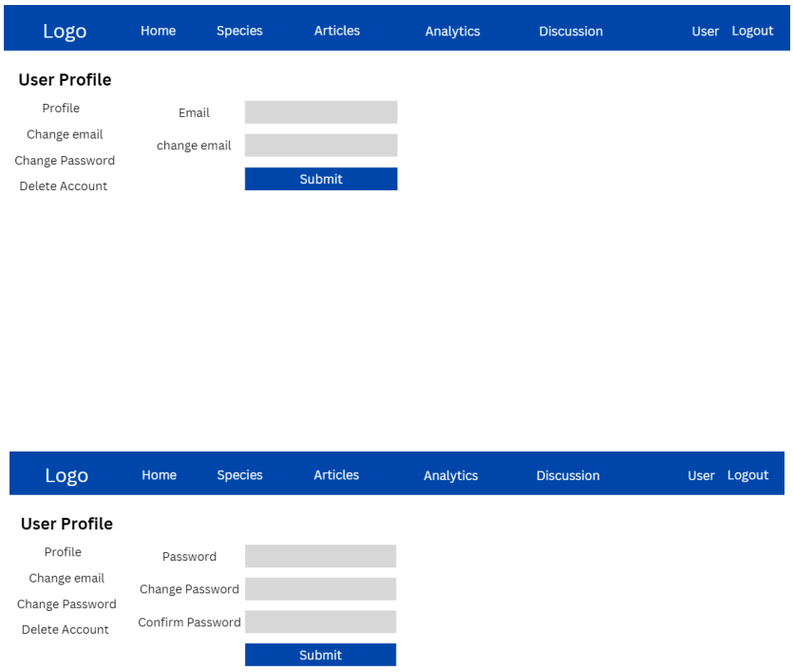


Figure 13. User Accounts

This wireframe presents the user profile of Marine Biodiversity Explorer where the user can edit his/her email , password or delete his/her account .

VIII. PROCEDURE

Step 1

Open Visual Studio 2022 and select “Create new Project”

Step 2

Select project template ASP.NET Core Web App (Model-View-Controller)

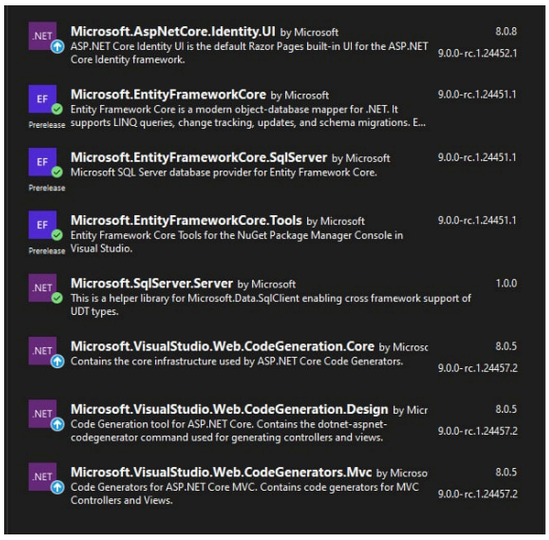
Step 3

After creating project navigate and select tools section then select NuGet Package Manager and Manage Packages for solution

Step 4

Search/Browse and install the following in Manage Packages for Solution

College of Informatics and Computing Sciences



check the boxes and and view latest version if it is on latest stable before installing

Step 5

after installing right click on the project name and select add then “new scaffolded item”then click on identity

Step 6

Find and Select “Account/Login, Account/Logout, Account/Register and Account/Reset Password”

Step 7

Click the three dot menu on the add identity section and go to Views then Shared folder and click Layout.cshtml then in data context class change name and set it to AppDbContext.cs and set User class to MarineUser.cs then click add

Step 8

Navigate Areas folder in right corner in Solution Explorer inside areas folder click Account folder and select Register.cshtml

Step 9

to run the program select layout.cshtml in the Shared folder in the right corner (Search solution explorer) to find layout.cshtml then press the green play button on the top corner of the application .

Step 10

write the highlighted part in Layout.cshtml to display login and register in the website

Layout.cshtml

```
<!DOCTYPE html>

<html lang="en">

<head>
```

```
    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>@ViewData["Title"] - MarinebiodiversityExplorer</title>

    <link rel="stylesheet" href="~/lib/bootstrap/dist/css/bootstrap.min.css" />

    <link rel="stylesheet" href="~/css/site.css" asp-append-version="true" />

    <link rel="stylesheet" href="~/MarinebiodiversityExplorer.styles.css" asp-append-version="true" />

</head>

<body>

    <header>

        <nav class="navbar navbar-expand-sm navbar-toggleable-sm navbar-light bg-white border-bottom
box-shadow mb-3">

            <div class="container-fluid">

                <a class="navbar-brand" asp-area="" asp-controller="Home"
asp-action="Index">MarinebiodiversityExplorer</a>

                <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target=".navbar-collapse" aria-controls="navbarSupportedContent"

                        aria-expanded="false" aria-label="Toggle navigation">

                    <span class="navbar-toggler-icon"></span>

                </button>

                <div class="navbar-collapse collapse d-sm-inline-flex justify-content-between">

                    <ul class="navbar-nav flex-grow-1">

                        <li class="nav-item">

                            <a class="nav-link text-dark" asp-area="" asp-controller="Home"
asp-action="Index">Home</a>

                        </li>

                        <li class="nav-item">

                            <a class="nav-link text-dark" asp-area="" asp-controller="Home"
asp-action="Privacy">Privacy</a>

                        </li>

                    </ul>

                </div>

            </div>
```

```
        </nav>

        <partial name="_Loginpartial"/>

    </header>

    <div class="container">

        <main role="main" class="pb-3">

            @RenderBody()

        </main>

    </div>


    <footer class="border-top footer text-muted">

        <div class="container">

            &copy; 2024 - MarinebiodiversityExplorer - <a asp-area="" asp-controller="Home"
asp-action="Privacy">Privacy</a>

        </div>

    </footer>

    <script src="~/lib/jquery/dist/jquery.min.js"></script>

    <script src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>

    <script src="~/js/site.js" asp-append-version="true"></script>

    @await RenderSectionAsync("Scripts", required: false)

</body>

</html>
```

Step 11

find and select C# Programs.cs and write the highlighted part below

```
using Microsoft.AspNetCore.Identity;

using Microsoft.EntityFrameworkCore;

using MarinebiodiversityExplorer.Areas.Identity.Data;

var builder = WebApplication.CreateBuilder(args);
```

```
var connectionString = builder.Configuration.GetConnectionString("AppDbContextConnection") ??
throw new InvalidOperationException("Connection string 'AppDbContextConnection' not found.");


builder.Services.AddDbContext<AppDbContext>(options => options.UseSqlServer(connectionString));


builder.Services.AddDefaultIdentity<MarineUser>(options => options.SignIn.RequireConfirmedAccount
= true).AddEntityFrameworkStores<AppDbContext>();


// Add services to the container.

builder.Services.AddControllersWithViews();


var app = builder.Build();


// Configure the HTTP request pipeline.

if (!app.Environment.IsDevelopment())

{

    app.UseExceptionHandler("/Home/Error");

    // The default HSTS value is 30 days. You may want to change this for production scenarios, see
https://aka.ms/aspnetcore-hsts.

    app.UseHsts();

}


app.UseHttpsRedirection();

app.UseStaticFiles();


app.UseRouting();


app.UseAuthorization();


app.MapControllerRoute(

    name: "default",
```

```
    pattern: "{controller=Home}/{action=Index}/{id?}");


app.MapRazorPages();

app.Run();
```

Step 12

in MarineUser.cs set property in public class SampleUser : IdentityUser

MarineUser.cs

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Identity;


namespace MarinebiodiversityExplorer.Areas.Identity.Data;


// Add profile data for application users by adding properties to the MarineUser class

public class MarineUser : IdentityUser

{

    public string FirstName { get; set; }

    public string LastName { get; set; }

}
```

Step 13

in AppDbContext.cs create a class to inherit identity to take user class then implement interface then inject builder.ApplyConfiguration(new ApplicationUserEntityConfiguration()); to protected override void then add property to the class

AppDbContext.cs

```
using MarinebiodiversityExplorer.Areas.Identity.Data;

using Microsoft.AspNetCore.Identity;
```

```
using Microsoft.AspNetCore.Identity.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore.Metadata.Builders;


namespace MarinebiodiversityExplorer.Areas.Identity.Data;


public class AppDbContext : IdentityDbContext<MarineUser>

{

    public AppDbContext(DbContextOptions<AppDbContext> options)

        : base(options)

    {

    }


    protected override void OnModelCreating(ModelBuilder builder)

    {

        base.OnModelCreating(builder);

        // Customize the ASP.NET Identity model and override the defaults if needed.

        // For example, you can rename the ASP.NET Identity table names and more.

        // Add your customizations after calling base.OnModelCreating(builder);

        builder.ApplyConfiguration(new ApplicationUserEntityConfiguration());

    }

}


public class ApplicationUserEntityConfiguration : IEntityTypeConfiguration<MarineUser>

{

    public void Configure(EntityTypeBuilder<MarineUser> builder)

    {

        builder.Property(x => x.FirstName).HasMaxLength(100);

        builder.Property(x => x.LastName).HasMaxLength(100);
```

```
    }

}
```

Step 14

open tools select nuget package manager then package manager console to migrate data

PM > Add-Migration InitialMigration

Step 15

go to appsettings.json to change server name , open sql server management studio and copy the server name of the device then return to visual studio , after changing the server name go to nuget package manager and select package manager console then write update-database

Step 16

go to tools then select connect to database and choose data source microsoft sql server

Step 17

go to sql configuration manager and create a database then , go to visual studio connect database to database name

Step 18

go to models and create a class then after creating a class , create properties inside it

Step 19

after creating properties go to data folder and select applicationDb context and inside it

type public DbSet<class name> class names { get; set;}

Step 20

go to tools and select nuget package manager then package manager console

step 21

type in console Add-migration initialmigrations after building type update-database then refresh the the database connected on Data connections in the left corner of the screen .Pa

Republic of the Philippines
BATANGAS STATE UNIVERSITY
The National Engineering University
Lipa Campus
A. Tanco Drive, Brgy. Marawoy, Lipa, Batangas, Philippines 4217
Tel Nos.: (+63 43) 980-0385; 980-0387; 980-0392 to 94 loc. 3130
E-mail Address: cics.lipa@g.batstate-u.edu.ph | Website Address: http://www.batstate-u.edu.ph

College of Informatics and Computing Sciences