North South University

Department

of

Electrical and Computer Engineering

Project: Bangladesh National Management Portal

Course Code: CSE311

Section : 12

Date of Submission: 02-12-2024

Submitted By: group-14

Name: Shakil Ahmed

ID: 222 1453 042

Name: Nael

ID: 2212254642

Bangladesh National Portal Management System

The **Bangladesh National Portal Management System (BNPMS)** is designed to consolidate public services and resources, providing citizens with an efficient way to request services and interact with government entities. The system will also empower government officials to manage requests and monitor service delivery, thereby improving transparency, efficiency, and accountability.

1. Project Overview

The BNPMS aims to serve as a centralized digital platform where:

- Citizens can request government services, track request statuses, and provide feedback.
- Government departments can manage service requests, monitor feedback, and oversee departmental operations.
- Government officials can access department-specific data, track performance metrics, and oversee service delivery.

2. Database Scope

The database is structured to support complex interactions between users, service requests, and government departments. It includes multiple types of entities and relationships to reflect real-world processes accurately.

Entities and Attributes

1. **User Types**:
    - **User**: A generalized entity containing basic attributes such as *User ID*, *Full Name*, *Username*, *Password*, *Email*, and *Notification Preferences*.
    - **Specialized User Types**:
        - **Citizen**: Specific attributes include *Citizen ID*, *Name*, *Date of Birth*, *Nationality*, *Marital Status*, *Occupation*, *Address* (present and permanent), and *Contact Information*. Derived attributes like *Age* are calculated from the date of birth.
        - **Expat**: Contains attributes relevant to foreign workers, such as *Visa Type*, *Work Permit Status*, *Expected Departure Date*, *Entry Date*, *Bank Account*, and *Origin*.
2. **NID_Card**:
    - Related to *Citizen*, it represents details from the national ID card, including *NID*, *Father's Name*, *Mother's Name*, *Date of Issue*, *Expiry Date*, *Blood Type*, *Place of Birth*, and *Signature*.

3. **Government Department**:
    - Government Official: Includes Official ID, Employment Type, Date of Appointment, Rank, Work Location, Supervisor, and Training Records. Officials manage or oversee services within departments.
    - Represents government departments with attributes such as *Department ID*, *Department Name*, *Founding Date*, *Location*, *Budget*, *Number of Employees*, *Contact Information*, and *Key Policies*.
    - Relationships like Under and Department_Head reflect hierarchical structures and management within government entities.
4. **Government Official:**
    - Includes *Official ID*, *Employment Type*, *Date of Appointment*, *Rank*, *Work Location*, *Supervisor*, and *Training Records*. Officials manage or oversee services within departments. There will be another specified field which will be called role and it will be Government official itself. A slected few persons will be elected as department head for certain period of time and this is implimented via role which we call Department_Head

5. **Services**:
    - Represents services provided by government departments. Attributes include *Service ID*, *Service Type*, *Service Description*, *Application Process*, *Priority Level*, *Documents Required*, and *Service History*.
6. **Service Request**:
    - Represents individual requests made by citizens for government services. Attributes include *Request ID*, *Request Status*, *Request Description*, *Supporting Evidence*, and the department responsible for handling the request.
    - Specialization into *Pending* and *Completed* allows for managing requests based on their current status:
        - **Pending Requests**: Includes attributes like *Submission Date*, *Last Updated Date*, *Department in Charge*, and *Follow-up Requirement*.
        - **Completed Requests**: Contains *Completion Date*, *Approval Status*, *Resolution Summary*, and *Final Processing*.
7. **Service Feedback**:
    - Allows citizens to provide feedback on services received. Attributes include *Feedback ID*, *Feedback Date*, *Comments*, *Rating*, and *Final Document Issued*.

8. **Notifications**:
    - Represents messages sent to users about service updates and important alerts, containing attributes like *Notification ID*, *Message*, *Notification Type*, and *Date Sent*.

Relationships and Constraints

1. **Hierarchies**:
    - The ISA relationship between *User* and specialized types (*Citizen*, *Government Official*, *Expat*) allows for role-based access and tailored functionality for different user types.
2. **Ternary Relationships**:
    - For example, the relationship among *Citizen, Expat* and *Service* enables detailed tracking of which service request is necessary for the user.
3. **Access Control**:
    - Roles and relationships (e.g., *Citizen*'s Can request for *Services*) define access permissions, ensuring that citizens can access their requests and government officials can oversee department-specific data.
4. **Cascading Actions**:
    - The database will enforce cascading updates and deletions. For example, if a department is removed, all associated service requests will be automatically deleted to maintain data integrity.

3. Implementation Scope

Database Design

1. **Keys and Constraints**:
    - Primary keys (e.g., *Citizen ID*, *Department ID*) and foreign keys will enforce data integrity.
    - Constraints will ensure valid data entry, such as mandatory fields for *Request Status* and *Service Type*.
2. **Composite and Multivalued Attributes**:
    - Attributes like *Name* composite.
    - Some attributes, like *Contact Information* for Citizen for example email, phone are multivalued to allow multiple entries.
3. **Derived Attributes**:
    - Certain attributes, like *Age*, will be derived from other data points, such as date of birth.

Stored Procedures and Functions

1. **Functions**:
    - Functions will calculate metrics like the total number of requests by a citizen or the average processing time for requests in each department.
2. **Procedures**:
    - Stored procedures will automate key tasks:
        - Assigning requests to government officials.
        - Updating request statuses based on progress.
        - Generating periodic reports for tracking department performance and citizen satisfaction.

Front-End Functionality

1. **User Interface and Interactions**:
    - The interface will support CRUD (Create, Read, Update, Delete) operations on major entities:
        - **Service Requests**: Citizens can submit new requests, view request statuses, update details, and delete outdated requests.
        - **Service Feedback**: Citizens can leave feedback on completed services, providing insights for improvement.
        - **Department Management**: Officials can manage department details, ensuring records are up-to-date.
2. **Notifications and Alerts**:
    - A notification system will keep citizens informed about the status of their requests and send important updates or alerts.
3. **Role-Based Access Control**:
    - Role-based views and permissions will allow citizens to access personal data, while officials have broader department-specific access, and administrators can oversee all data.

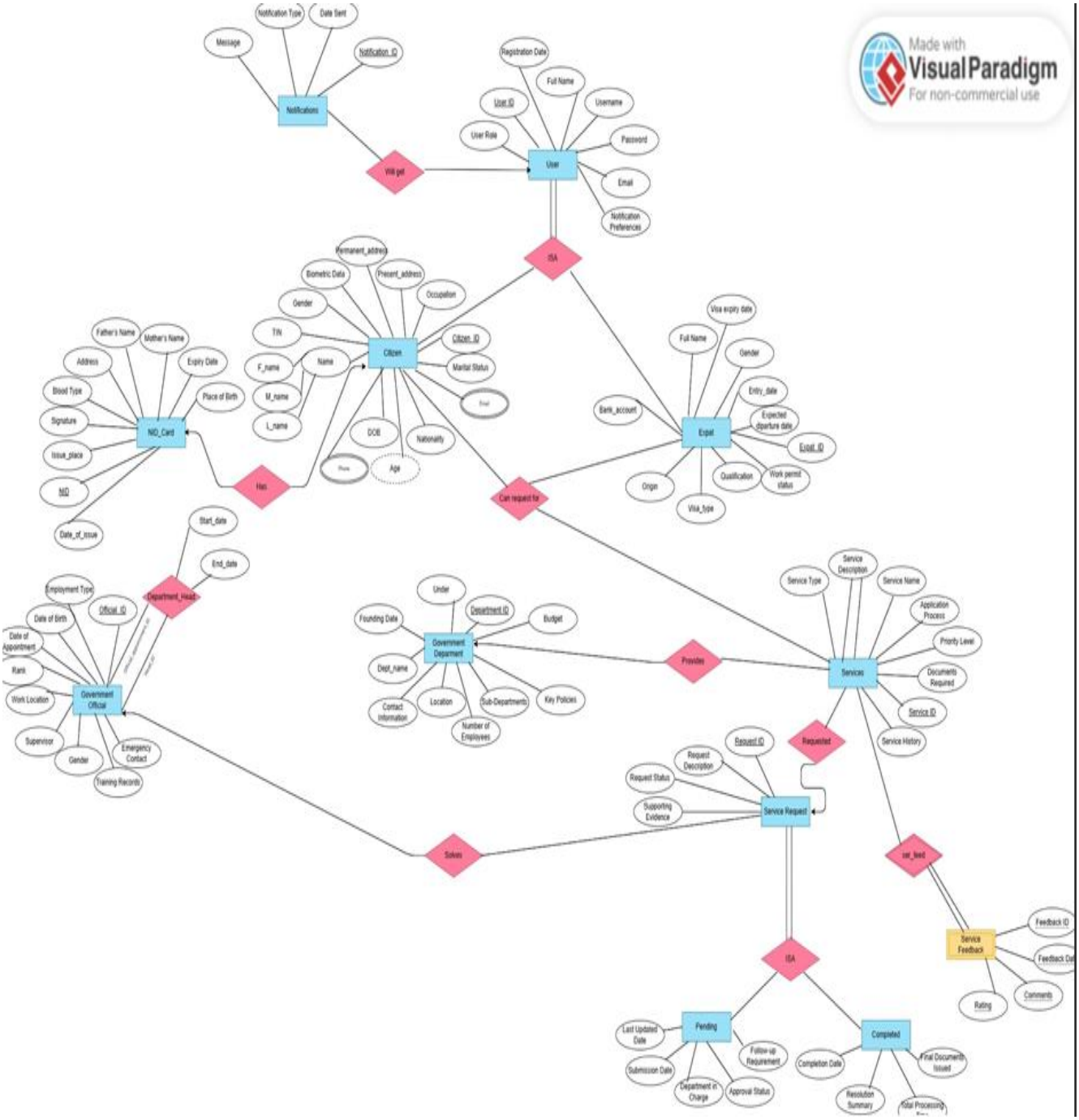
4. Expected Outcomes

This scope defines a robust and scalable BNPMS that will support:

- **Enhanced Citizen Engagement**: Through service request tracking, feedback collection, and notifications, citizens will feel more connected and empowered.
- **Improved Efficiency for Government Officials**: With streamlined data and automated tasks, officials can manage service requests more effectively.

- **Increased Transparency**: By enabling citizens to view request statuses and provide feedback, the system will promote government accountability.

ERD Diagram:


Notification Type
Date Sent
Message
Notification_ID
Notifications
Will get
Registration Date
Full Name
User ID
Username
User Role
Password
User
Email
Notification Preferences
ISA
Permanent_address
Biometric Data
Present_address
Occupation
Gender
TIN
Citizen_ID
Father's Name
Mother's Name
Address
Expiry Date
F_name
Name
Marital Status
Blood Type
Place of Birth
M_name
Citizen
Signature
L_name
DOB
Nationality
Email
Phone
Age
NID_Card
Issue_place
NID
Has
Date_of_issue
Visa expiry date
Full Name
Gender
Entry_date
Bank_account
Expat
Expected departure date
Expat_ID
Origin
Qualification
Work permit status
Visa_type
Can request for
Start_date
End_date
Department_Head
Employment Type
Official_ID
Date of Birth
Date of Appointment
Rank
Work Location
Government Official
Supervisor
Gender
Emergency Contact
Training Records
Under
Department ID
Budget
Founding Date
Government Deparment
Dept_name
Contact Information
Location
Sub-Departments
Key Policies
Number of Employees
Provides
Service Description
Service Type
Service Name
Application Process
Priority Level
Services
Documents Required
Service ID
Service History
Requested
Request ID
Request Description
Request Status
Supporting Evidence
Service Request
Solves
set_feed
Service Feedback
Feedback ID
Feedback Date
Comments
Rating
ISA
Pending
Last Updated Date
Submission Date
Follow-up Requirement
Department in Charge
Approval Status
Completed
Completion Date
Final Documents Issued
Resolution Summary
Total Processing

Relational schema-

Citizen (**Citizen_ID**, F_name, M_name, L_name, Gender, DOB, Nationality, Marital_Status, Occupation, Present_address, Permanent_address, Biometric_Data, TIN)

Expat (**Expat_ID**, Full_Name, Gender, Origin, Visa_type, Visa_expiry_date, Qualification, Entry_date, Expected_departure_date, Work_permit_status, Bank_account)

NID_Card (**NID**, Citizen_ID, Fathers_Name, Mothers_Name, Address, Blood_Type, Signature, Place_of_Birth, Issue_place, Expiry_date, Date_of_issue)

User (**User_ID**, Full_Name, Username, Password, Email, User_Role, Notification_Preferences, Registration_Date)

Government_Official (**Official_ID**, Full_Name, Date_of_Birth, Employment_Type, Date_of_Appointment, Rank, Work_Location, Supervisor, Gender, Emergency_Contact, Training_Records)

Government_Department (**Department_ID**, Dept_name, Founding_Date, Location, Contact_Information, Budget, Key_Policies, Number_of_Employees, Sub_Department)

Services (**Service_ID**, Department_ID, Request_ID, Service_Name, Service_Type, Service_Description, Application_Process, Priority_Level, Documents_Required, Service_History)

Notifications (**Notification_ID**, User_ID, Message, Notification_Type, Date_Sent)

Service Request (**Request_ID**, Official_ID, Request_Status, Request_Description, Supporting_Evidence)

Pending (**Request_ID**, Last_Updated_Date, Submission_Date, Department_in_Charge, Approval_Status, Follow_up_Requirement)

Completed (**Request_ID**, Completion_Date, Resolution_Summary, Final_Documents_Issued, Total_Processing_Time)

Service Feedback (**Service_ID,Feedback_ID, Rating, Comments, Feedback_Date)**

Department_Head (**Head_ID, Official_appointment_ID, Start_date, End_date)**

Can request for (**Service_ID, Citizen_ID, Expat_ID)**

Citizen_email**(Citizen_ID, citizen_email**)

Citizen_phone **(Citizen_ID, citizen_phone**)

SQL DDL for the Relation Schema:

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";


-- Database: `bdportal`


-- Table structure for table `citizen`


CREATE TABLE `citizen` (
  `CitizenID` int(12) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `UserID` int(12) NOT NULL,
  `FullName` varchar(100) DEFAULT NULL,
  `DateOfBirth` date DEFAULT NULL,
  `Nationality` varchar(50) DEFAULT NULL,
  `MaritalStatus` varchar(20) DEFAULT NULL,
  `Occupation` varchar(50) DEFAULT NULL,
  `addressPresent` varchar(255) DEFAULT NULL,
  `addressPermanent` varchar(255) DEFAULT NULL,
  `ContactInfo` varchar(100) DEFAULT NULL,
  `Age` int(10) DEFAULT NULL,
  `TIN` int(20) DEFAULT NULL,

  FOREIGN KEY (`UserID`) REFERENCES `users` (`UserID`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```
--
-- Triggers `citizen`
--
DELIMITER $$
CREATE TRIGGER `set_age_before_insert` BEFORE INSERT ON `citizen` FOR EACH ROW
BEGIN
  SET NEW.Age = TIMESTAMPDIFF(YEAR, NEW.DateOfBirth, CURDATE());
END
$$
DELIMITER ;
DELIMITER $$
CREATE TRIGGER `set_age_before_update` BEFORE UPDATE ON `citizen` FOR EACH ROW
BEGIN
  SET NEW.Age = TIMESTAMPDIFF(YEAR, NEW.DateOfBirth, CURDATE());
END
$$
DELIMITER ;


-- --------------------------------------------------------


--
-- Table structure for table `completedrequest`
--


CREATE TABLE `completedrequest` (
 `RequestID` int(11) NOT NULL,
 `CompletionDate` date DEFAULT NULL,
 `ApprovalStatus` varchar(50) DEFAULT NULL,
 `ResolutionSummary` text DEFAULT NULL,
```

```
  `FinalProcessing` text DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--
-- Table structure for table `expat`
--

CREATE TABLE `expat` (
  `ExpatID` int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `UserID` int(12) NOT NULL,
  `VisaType` varchar(50) DEFAULT NULL,
  `WorkPermitStatus` varchar(50) DEFAULT NULL,
  `ExpectedDepartureDate` date DEFAULT NULL,
  `EntryDate` date DEFAULT NULL,
  `BankAccount` varchar(50) DEFAULT NULL,
  `Origin` varchar(50) DEFAULT NULL,
  `PassportNumber` varchar(50) DEFAULT NULL,

  FOREIGN KEY (`UserID`) REFERENCES `users` (`UserID`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------
-- Table structure for table `users`
--

CREATE TABLE `users` (
  `UserID` int(12) NOT NULL AUTO_INCREMENT PRIMARY KEY,
```

```
  `Username` varchar(50) NOT NULL,
  `Password` varchar(255) NOT NULL,
  `Email` varchar(100) DEFAULT NULL,
  `NotificationPreferences` varchar(50) DEFAULT NULL,
  `type` varchar(50) DEFAULT NULL,
  `date_registered` DATETIME DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `users`
--

INSERT INTO `users` (`UserID`, `FullName`, `Username`, `Password`, `Email`,
`NotificationPreferences`) VALUES
(1, 'John Doe', 'johndoe', 'johndoe', 'johndoe@mail.com', NULL),
(2, 'Jane Doe', 'janedoe', 'janedoe', 'janedoe@mail.com', NULL),
(3, 'Alice', 'alice', 'alice', 'alice@mail.com', NULL),
(4, 'Bob', 'bob', 'bob', 'bob@mail.com', NULL),
(5, 'Charlie', 'charlie', 'charlie', 'charlie@mail.com', NULL),
(6, 'David', 'david', 'david', 'david@mail.com', NULL);

--
--
-- Table structure for table `governmentdepartment`
--

CREATE TABLE `governmentofficial` (
  `OfficialID` int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `Username` varchar(50) NOT NULL UNIQUE,
```

```
  `FullName` varchar(100) NOT NULL,
  `Password` varchar(255) NOT NULL,
  `EmploymentType` varchar(50) DEFAULT NULL,
  `DateOfAppointment` date DEFAULT NULL,
  `WorkLocation` varchar(100) DEFAULT NULL,
  `Supervisor` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

CREATE TABLE `admin` (
  `Username` varchar(50) NOT NULL UNIQUE,
  `FullName` varchar(100) NOT NULL,
  `Password` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `department` (
  `DepartmentID` int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `DepartmentName` varchar(100) NOT NULL,
  `FoundingDate` date DEFAULT NULL,
  `Location` varchar(100) DEFAULT NULL,
  `Budget` decimal(15,2) DEFAULT NULL,
  `NumberOfEmployees` int(11) DEFAULT NULL,
  `ContactInfo` varchar(100) DEFAULT NULL,
  `KeyPolicies` text DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--
```

```
--
-- Table structure for table `nid_card`
--

CREATE TABLE `nid_card` (
  `NID` int(11) NOT NULL,
  `CitizenID` int(12) NOT NULL,
  `FathersName` varchar(100) DEFAULT NULL,
  `MothersName` varchar(100) DEFAULT NULL,
  `DateOfIssue` date DEFAULT NULL,
  `ExpiryDate` date DEFAULT NULL,
  `BloodType` char(3) DEFAULT NULL,
  `PlaceOfBirth` varchar(100) DEFAULT NULL,
  `Signature` blob DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--
-- Table structure for table `notifications`
--

CREATE TABLE `notifications` (
  `NotificationID` int(11) NOT NULL,
  `UserID` int(12) NOT NULL,
  `Message` text DEFAULT NULL,
  `NotificationType` varchar(50) DEFAULT NULL,
  `DateSent` date DEFAULT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--
-- Table structure for table `pendingrequest`
--

CREATE TABLE `pendingrequest` (
  `RequestID` int(11) NOT NULL,
  `SubmissionDate` date DEFAULT NULL,
  `LastUpdatedDate` date DEFAULT NULL,
  `DepartmentInCharge` int(11) DEFAULT NULL,
  `FollowUpRequirement` tinyint(1) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--
-- Table structure for table `servicefeedback`
--

CREATE TABLE `servicefeedback` (
  `FeedbackID` int(11) NOT NULL,
  `RequestID` int(11) NOT NULL,
  `FeedbackDate` date DEFAULT NULL,
  `Comments` text DEFAULT NULL,
  `Rating` int(11) DEFAULT NULL CHECK (`Rating` between 1 and 5),
  `FinalDocumentIssued` text DEFAULT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--
-- Table structure for table `servicerequest`
--

CREATE TABLE `servicerequest` (
  `RequestID` int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `CitizenID` int(12) NOT NULL,
  `ServiceID` int(11) DEFAULT NULL,
  `RequestStatus` varchar(50) NOT NULL DEFAULT 'Pending',
  `RequestDescription` text DEFAULT NULL,
  `SupportingEvidence` text DEFAULT NULL,

  FOREIGN KEY (`CitizenID`) REFERENCES `citizen` (`CitizenID`) ON DELETE CASCADE,
  FOREIGN KEY (`ServiceID`) REFERENCES `services` (`ServiceID`) ON DELETE SET NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--
-- Table structure for table `services`
--

CREATE TABLE `services` (
  `ServiceID` int(11) AUTO_INCREMENT PRIMARY KEY,
  `ServiceType` varchar(100) DEFAULT NULL,
```

```
  `ServiceDescription` text DEFAULT NULL,
  `ApplicationProcess` text DEFAULT NULL,
  `PriorityLevel` varchar(20) DEFAULT NULL,
  `DocumentsRequired` text DEFAULT NULL,
  `ServiceHistory` text DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

-- --------------------------------------------------------

--

-- Indexes for dumped tables
--

--
-- Indexes for table `citizen`
--
ALTER TABLE `citizen`
  ADD PRIMARY KEY (`CitizenID`),
  ADD UNIQUE KEY `UserID` (`UserID`);

--
-- Indexes for table `completedrequest`
--
ALTER TABLE `completedrequest`
  ADD PRIMARY KEY (`RequestID`);

--
-- Indexes for table `expat`
```

```
--
ALTER TABLE `expat`
 ADD PRIMARY KEY (`ExpatID`),
 ADD UNIQUE KEY `UserID` (`UserID`);

--
-- Indexes for table `governmentdepartment`
--
ALTER TABLE `governmentdepartment`
 ADD PRIMARY KEY (`DepartmentID`);

--
-- Indexes for table `governmentofficial`
--
ALTER TABLE `governmentofficial`
 ADD PRIMARY KEY (`OfficialID`),
 ADD UNIQUE KEY `UserID` (`UserID`),
 ADD KEY `Supervisor` (`Supervisor`);

--
-- Indexes for table `nid_card`
--
ALTER TABLE `nid_card`
 ADD PRIMARY KEY (`NID`),
 ADD UNIQUE KEY `CitizenID` (`CitizenID`);

--
-- Indexes for table `notifications`
--
```

```
ALTER TABLE `notifications`
 ADD PRIMARY KEY (`NotificationID`),
 ADD KEY `UserID` (`UserID`);

--
-- Indexes for table `pendingrequest`
--
ALTER TABLE `pendingrequest`
 ADD PRIMARY KEY (`RequestID`),
 ADD KEY `DepartmentInCharge` (`DepartmentInCharge`);

--
-- Indexes for table `servicefeedback`
--
ALTER TABLE `servicefeedback`
 ADD PRIMARY KEY (`FeedbackID`),
 ADD KEY `RequestID` (`RequestID`);

--
-- Indexes for table `servicerequest`
--
ALTER TABLE `servicerequest`
 ADD PRIMARY KEY (`RequestID`),
 ADD KEY `CitizenID` (`CitizenID`),
 ADD KEY `ServiceID` (`ServiceID`);

--
-- Indexes for table `services`
--
```

```
ALTER TABLE `services`
 ADD PRIMARY KEY (`ServiceID`);


--
-- Indexes for table `users`
--
ALTER TABLE `users`
 ADD PRIMARY KEY (`UserID`),
 ADD UNIQUE KEY `Username` (`Username`),
 ADD UNIQUE KEY `Email` (`Email`);


--
-- AUTO_INCREMENT for dumped tables
--


--
-- AUTO_INCREMENT for table `users`
--
ALTER TABLE `users`
 MODIFY `UserID` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;


--
-- Constraints for dumped tables
--


--
-- Constraints for table `citizen`
--
ALTER TABLE `citizen`
```

```
 ADD CONSTRAINT `citizen_ibfk_1` FOREIGN KEY (`UserID`) REFERENCES `users`
(`UserID`);


--

-- Constraints for table `completedrequest`

--

ALTER TABLE `completedrequest`

 ADD CONSTRAINT `completedrequest_ibfk_1` FOREIGN KEY (`RequestID`) REFERENCES
`servicerequest` (`RequestID`);


--

-- Constraints for table `expat`

--

ALTER TABLE `expat`

 ADD CONSTRAINT `expat_ibfk_1` FOREIGN KEY (`UserID`) REFERENCES `users`
(`UserID`);


--

-- Constraints for table `governmentofficial`

--

ALTER TABLE `governmentofficial`

 ADD CONSTRAINT `governmentofficial_ibfk_1` FOREIGN KEY (`UserID`) REFERENCES
`users` (`UserID`),

 ADD CONSTRAINT `governmentofficial_ibfk_2` FOREIGN KEY (`Supervisor`) REFERENCES
`governmentofficial` (`OfficialID`);


--

-- Constraints for table `nid_card`

--

ALTER TABLE `nid_card`
```

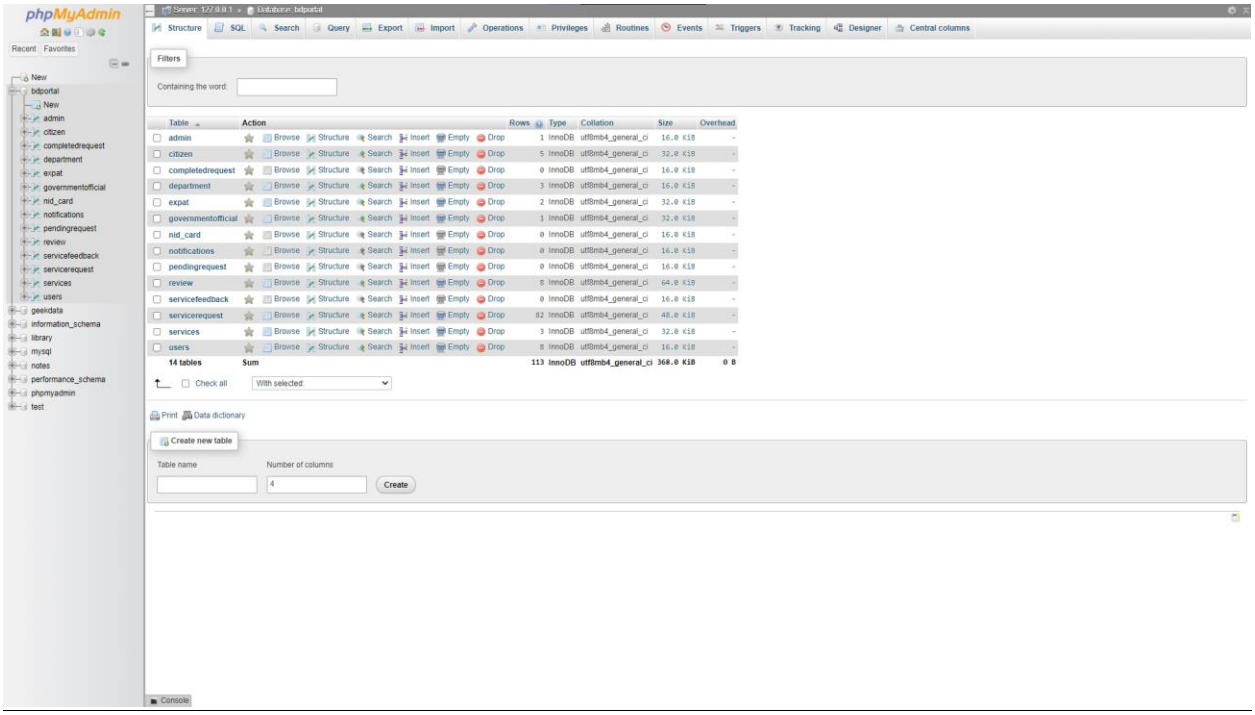
```
 ADD CONSTRAINT `nid_card_ibfk_1` FOREIGN KEY (`CitizenID`) REFERENCES `citizen`
(`CitizenID`);


--

-- Constraints for table `notifications`

--

ALTER TABLE `notifications`

 ADD CONSTRAINT `notifications_ibfk_1` FOREIGN KEY (`UserID`) REFERENCES `users`
(`UserID`);


--

-- Constraints for table `pendingrequest`

--

ALTER TABLE `pendingrequest`

 ADD CONSTRAINT `pendingrequest_ibfk_1` FOREIGN KEY (`RequestID`) REFERENCES
`servicerequest` (`RequestID`),

 ADD CONSTRAINT `pendingrequest_ibfk_2` FOREIGN KEY (`DepartmentInCharge`)
REFERENCES `governmentdepartment` (`DepartmentID`);


--

-- Constraints for table `servicefeedback`

--

ALTER TABLE `servicefeedback`

 ADD CONSTRAINT `servicefeedback_ibfk_1` FOREIGN KEY (`RequestID`) REFERENCES
`servicerequest` (`RequestID`);


--

-- Constraints for table `servicerequest`

--

ALTER TABLE `servicerequest`
```

ADD CONSTRAINT `servicerequest_ibfk_1` FOREIGN KEY (`CitizenID`) REFERENCES `citizen` (`CitizenID`),

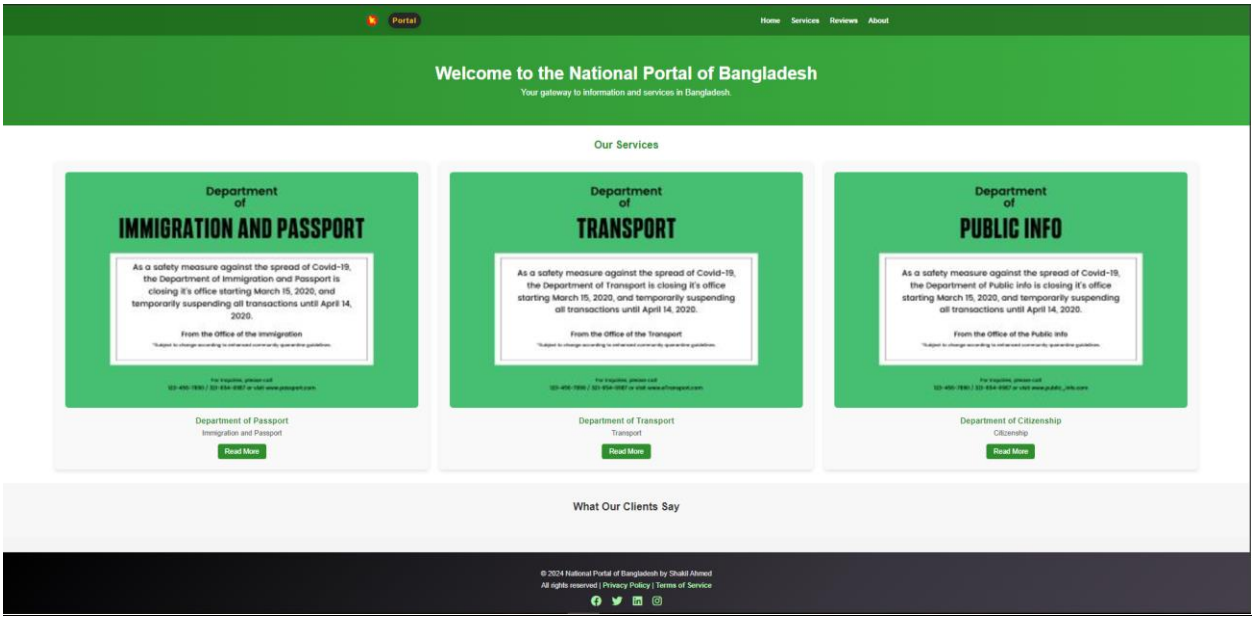
ADD CONSTRAINT `servicerequest_ibfk_2` FOREIGN KEY (`ServiceID`) REFERENCES `services` (`ServiceID`);

COMMIT;

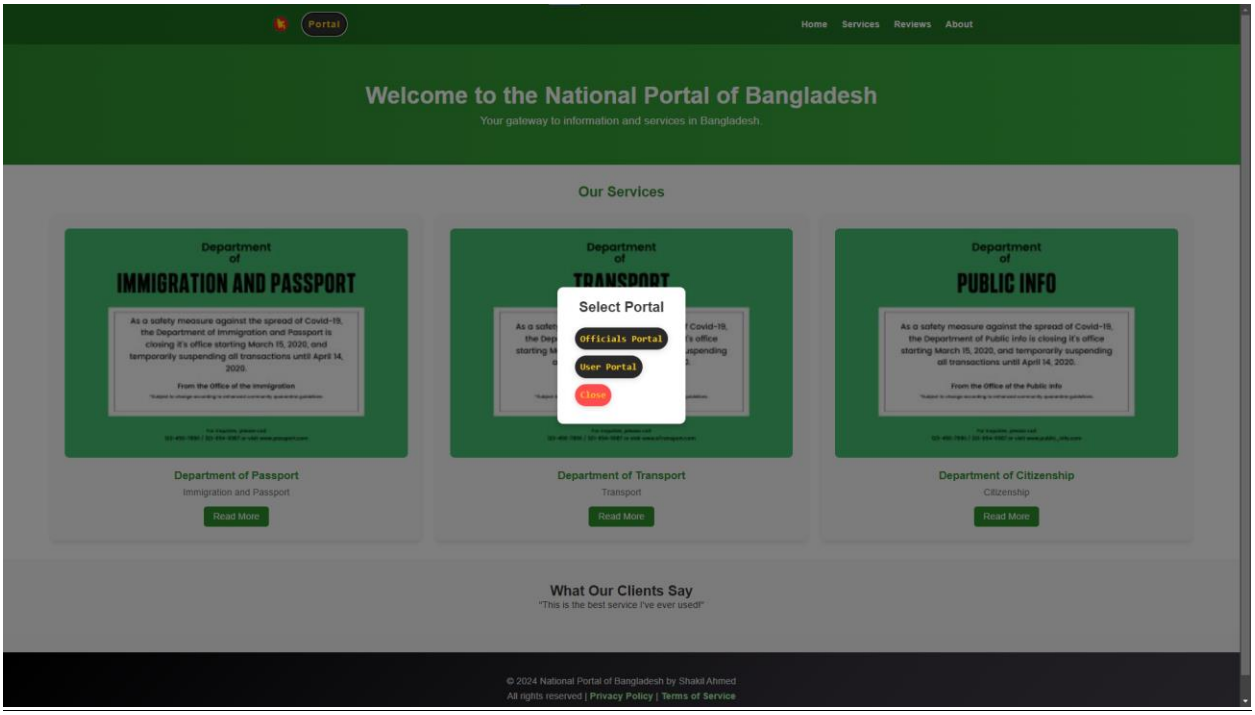
DataBase:


phpMyAdmin
Recent Favorites
New
bdportal
New
admin
citizen
completedrequest
department
expat
governmentofficial
nid_card
notifications
pendingrequest
review
servicefeedback
servicerequest
services
users
geekdata
information_schema
library
mysql
notes
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: bdportal
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer Central columns
Filters
Containing the word:
Table Action Rows Type Collation Size Overhead
admin Browse Structure Search Insert Empty Drop 1 InnoDB utf8mb4_general_ci 16.0 KiB -
citizen Browse Structure Search Insert Empty Drop 5 InnoDB utf8mb4_general_ci 32.0 KiB -
completedrequest Browse Structure Search Insert Empty Drop 0 InnoDB utf8mb4_general_ci 16.0 KiB -
department Browse Structure Search Insert Empty Drop 3 InnoDB utf8mb4_general_ci 16.0 KiB -
expat Browse Structure Search Insert Empty Drop 2 InnoDB utf8mb4_general_ci 32.0 KiB -
governmentofficial Browse Structure Search Insert Empty Drop 1 InnoDB utf8mb4_general_ci 32.0 KiB -
nid_card Browse Structure Search Insert Empty Drop 0 InnoDB utf8mb4_general_ci 16.0 KiB -
notifications Browse Structure Search Insert Empty Drop 0 InnoDB utf8mb4_general_ci 16.0 KiB -
pendingrequest Browse Structure Search Insert Empty Drop 0 InnoDB utf8mb4_general_ci 16.0 KiB -
review Browse Structure Search Insert Empty Drop 8 InnoDB utf8mb4_general_ci 64.0 KiB -
servicefeedback Browse Structure Search Insert Empty Drop 0 InnoDB utf8mb4_general_ci 16.0 KiB -
servicerequest Browse Structure Search Insert Empty Drop 82 InnoDB utf8mb4_general_ci 48.0 KiB -
services Browse Structure Search Insert Empty Drop 3 InnoDB utf8mb4_general_ci 32.0 KiB -
users Browse Structure Search Insert Empty Drop 8 InnoDB utf8mb4_general_ci 16.0 KiB -
14 tables Sum 113 InnoDB utf8mb4_general_ci 368.0 KiB 0 B
Check all With selected:
Print Data dictionary
Create new table
Table name
Number of columns
4
Create
Console

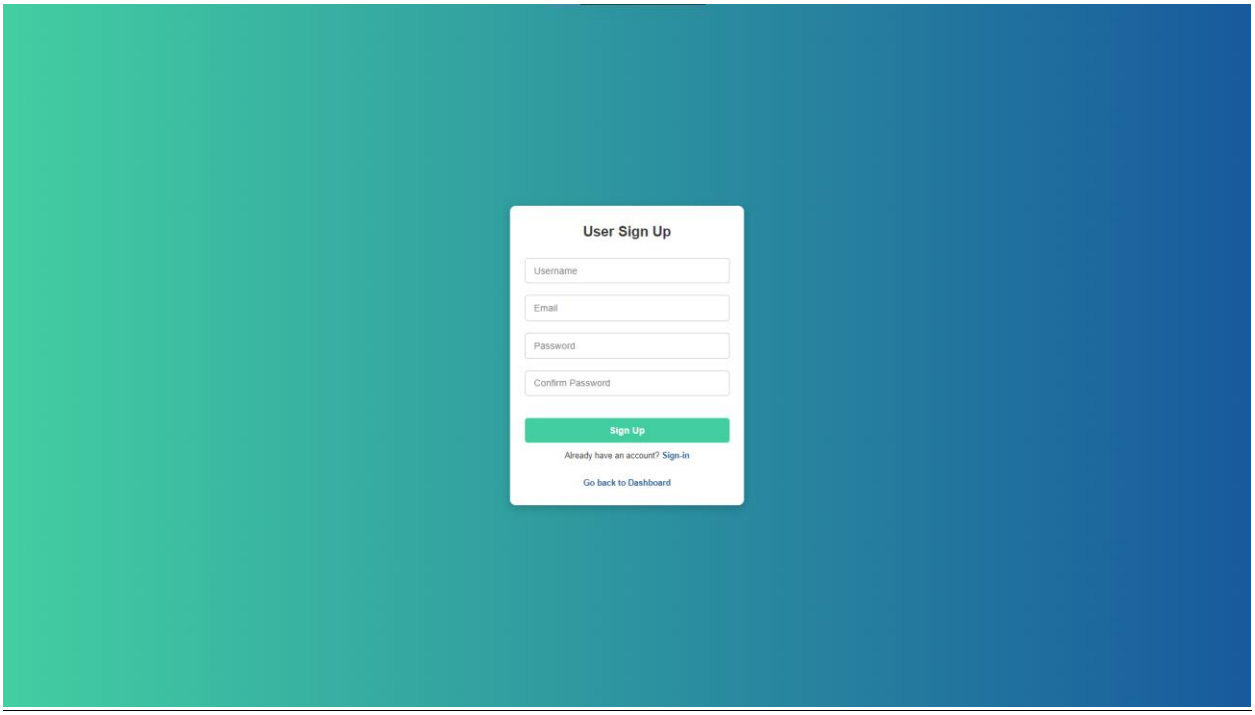
Dashboard :



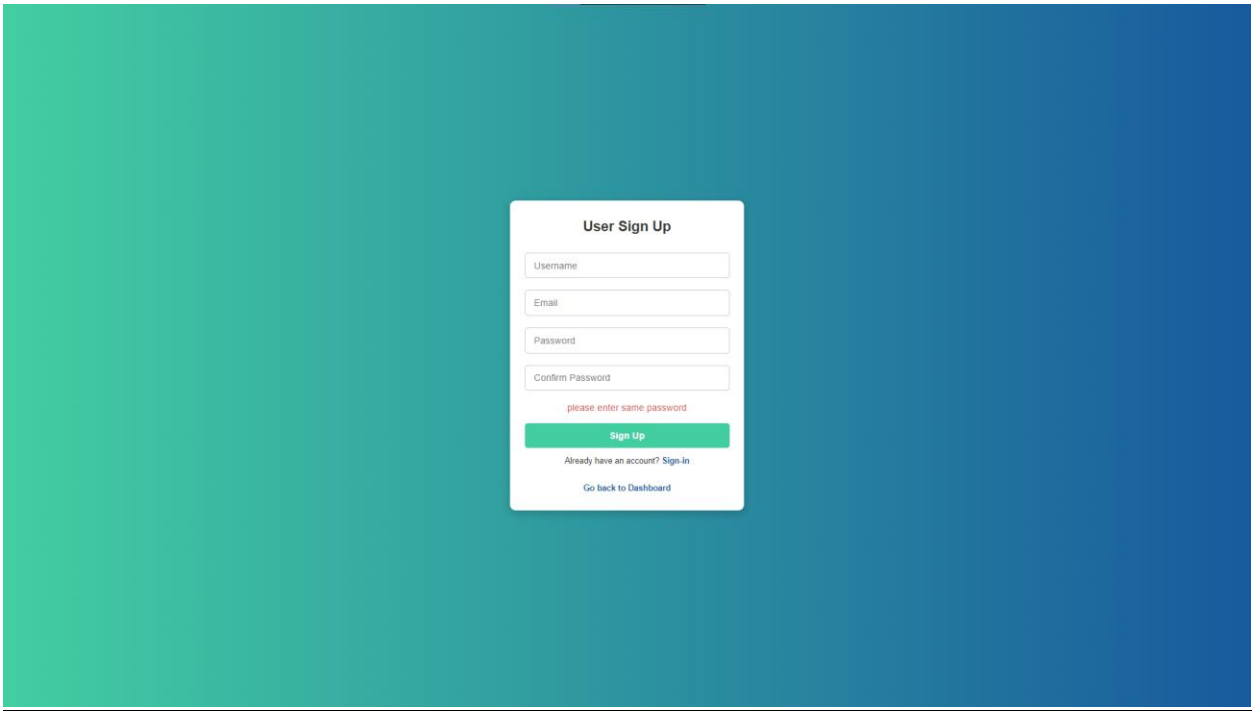
Portal Modal offers different types of Users Portal:



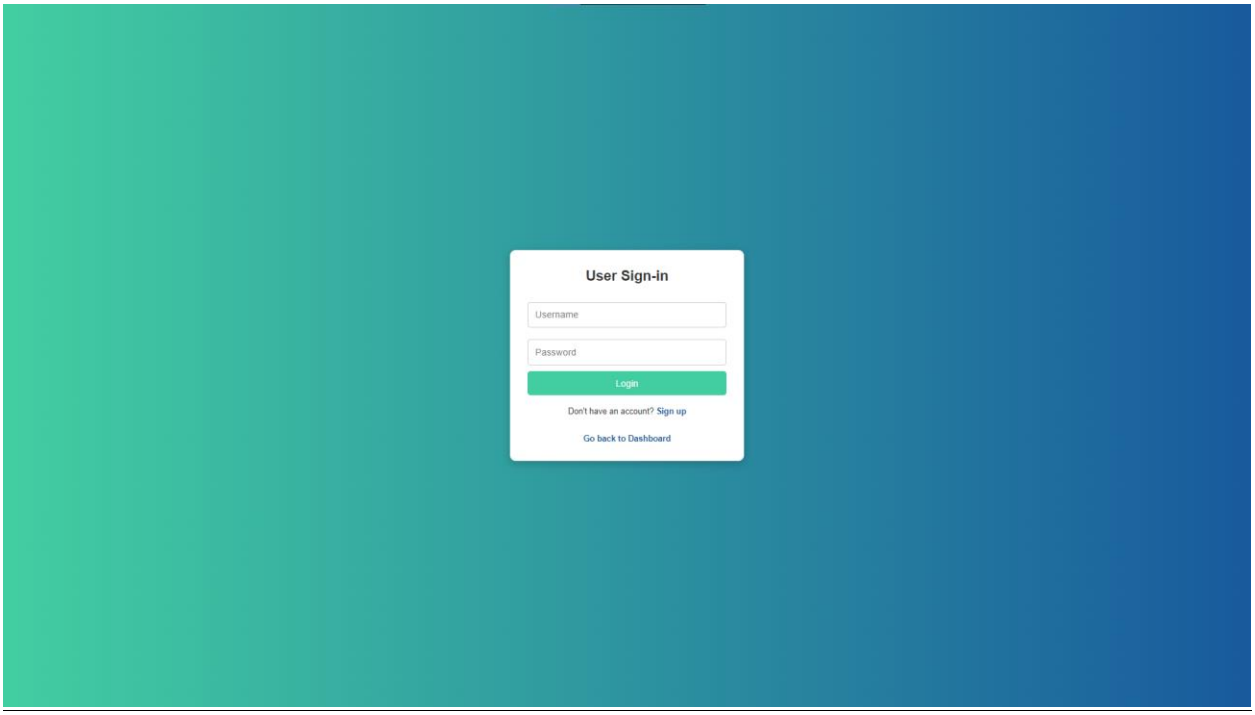
User Sign Up (Sign Up):



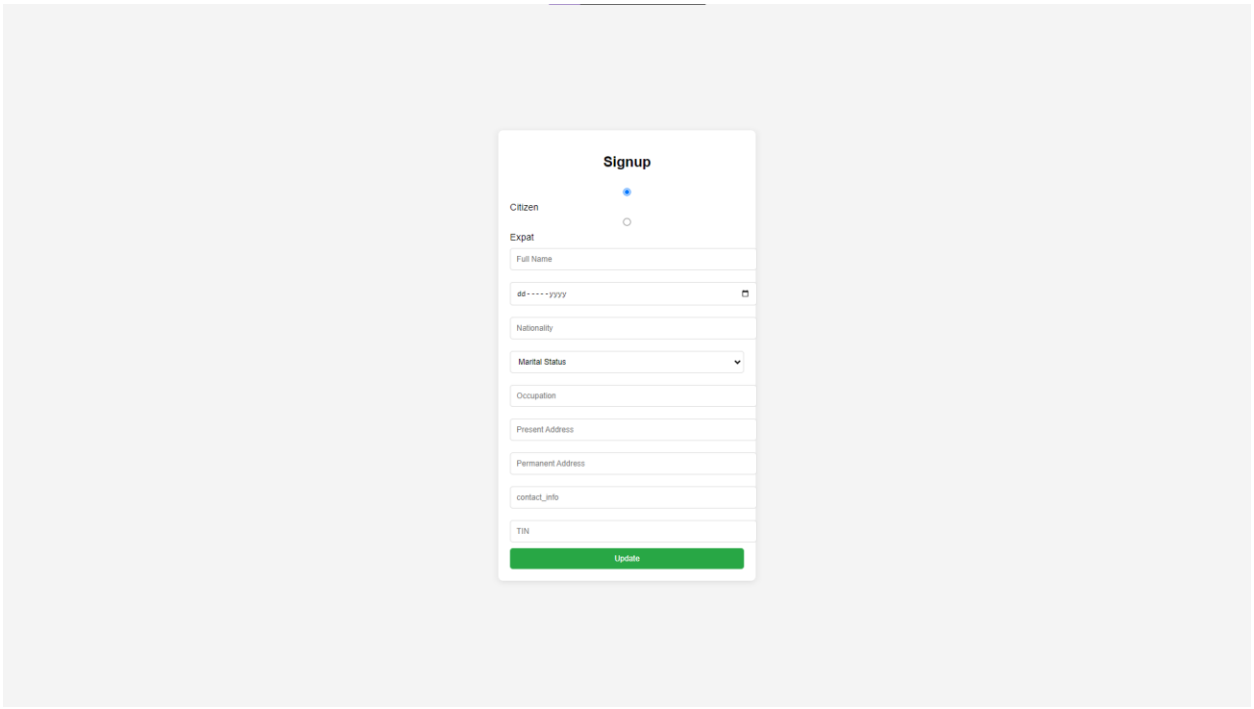
Error handling (e.g. Password and Confirm Password should match):



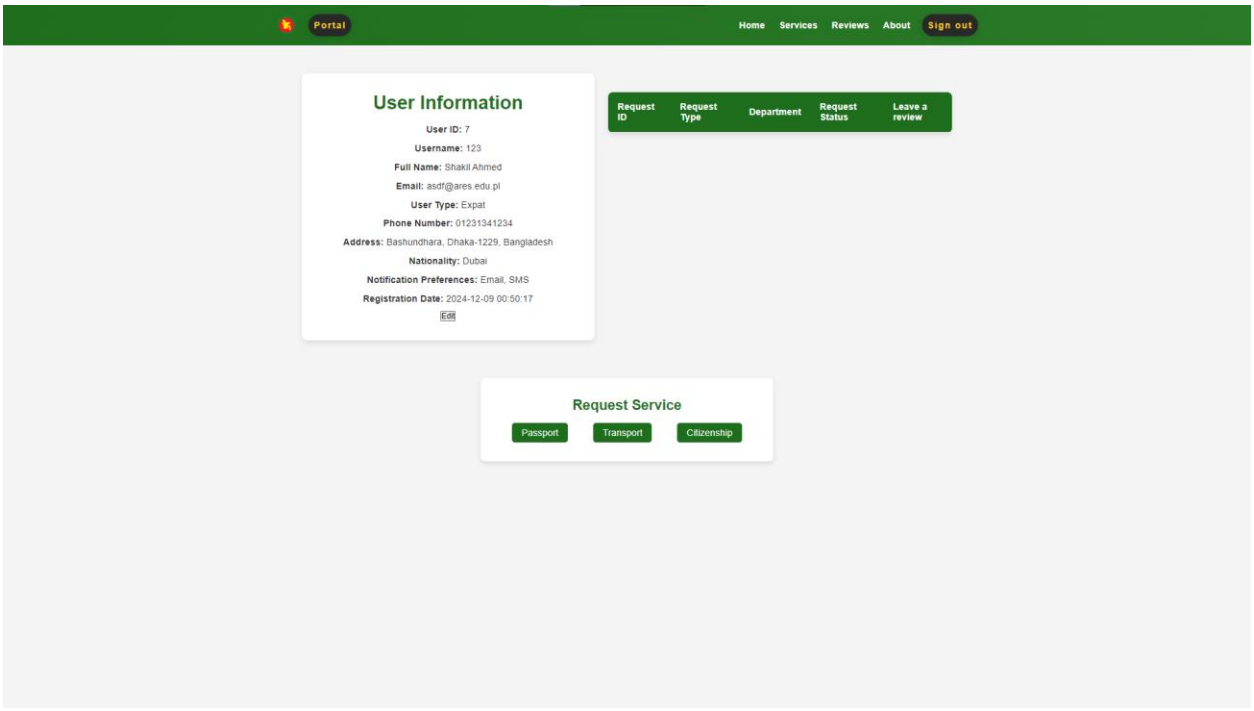
User Sign in(Upon Succesful Sign up):



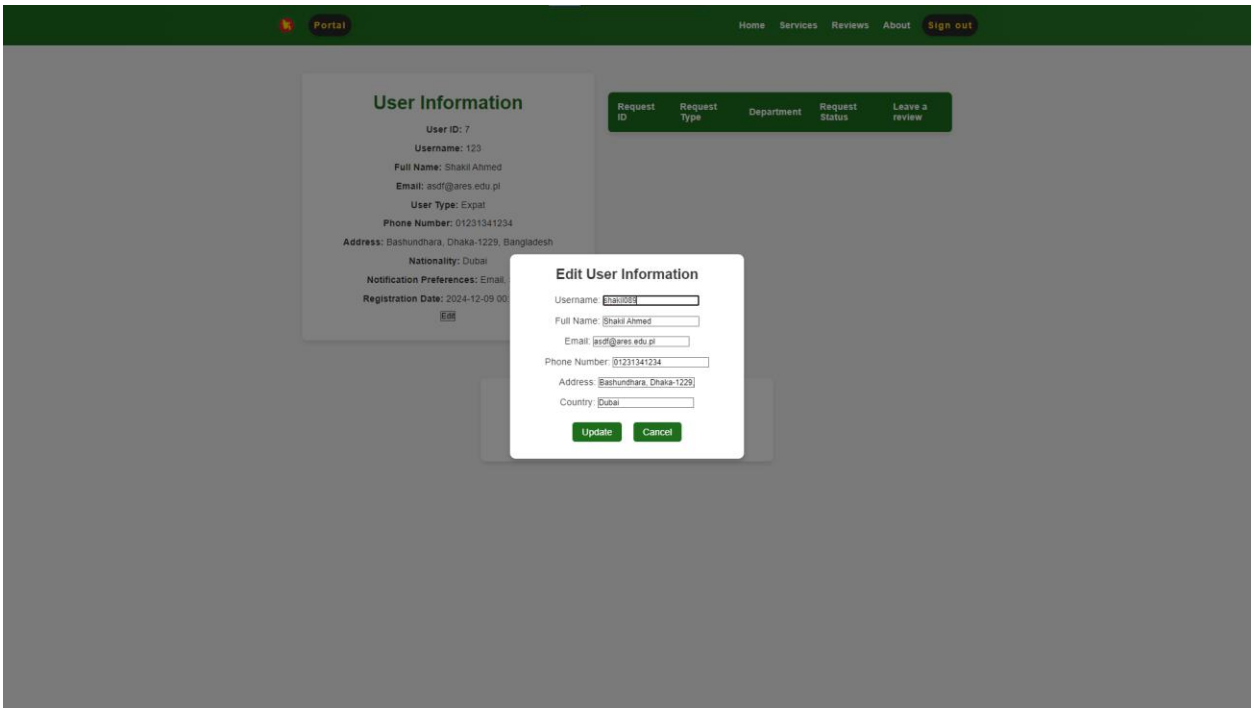
User info Store(Distinguishes if User is a Citizen or an Expat):



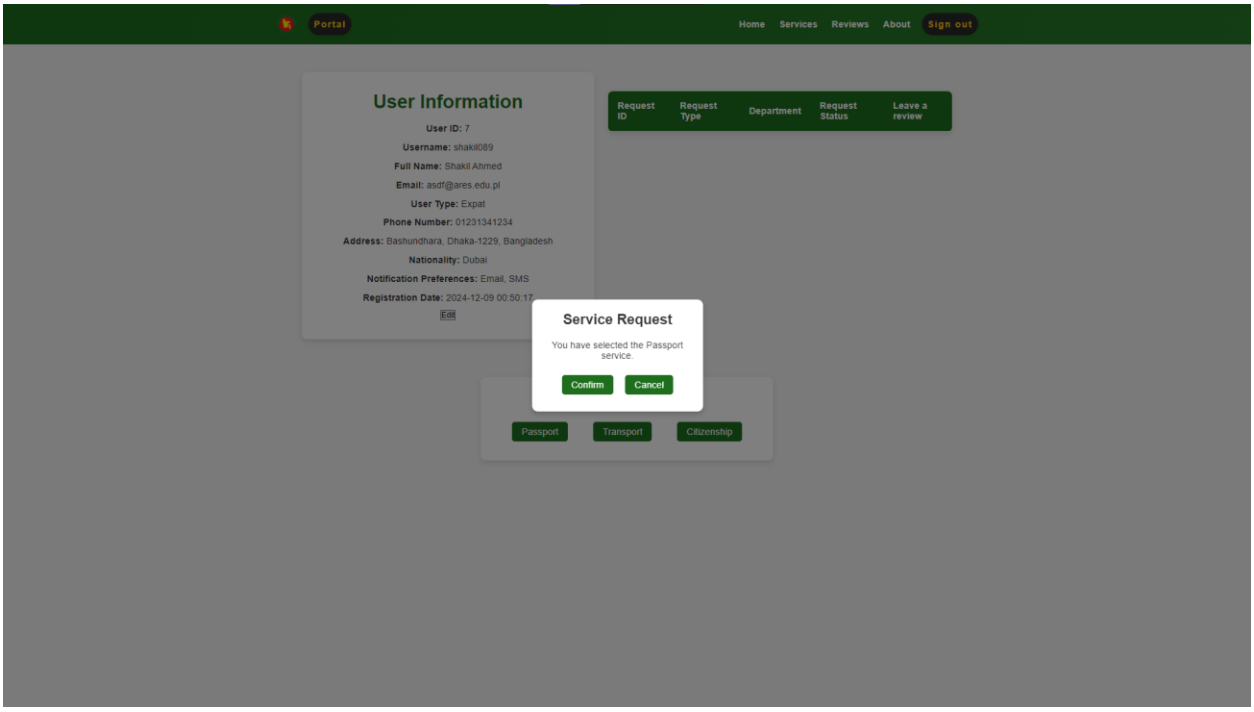
User DashBoard:



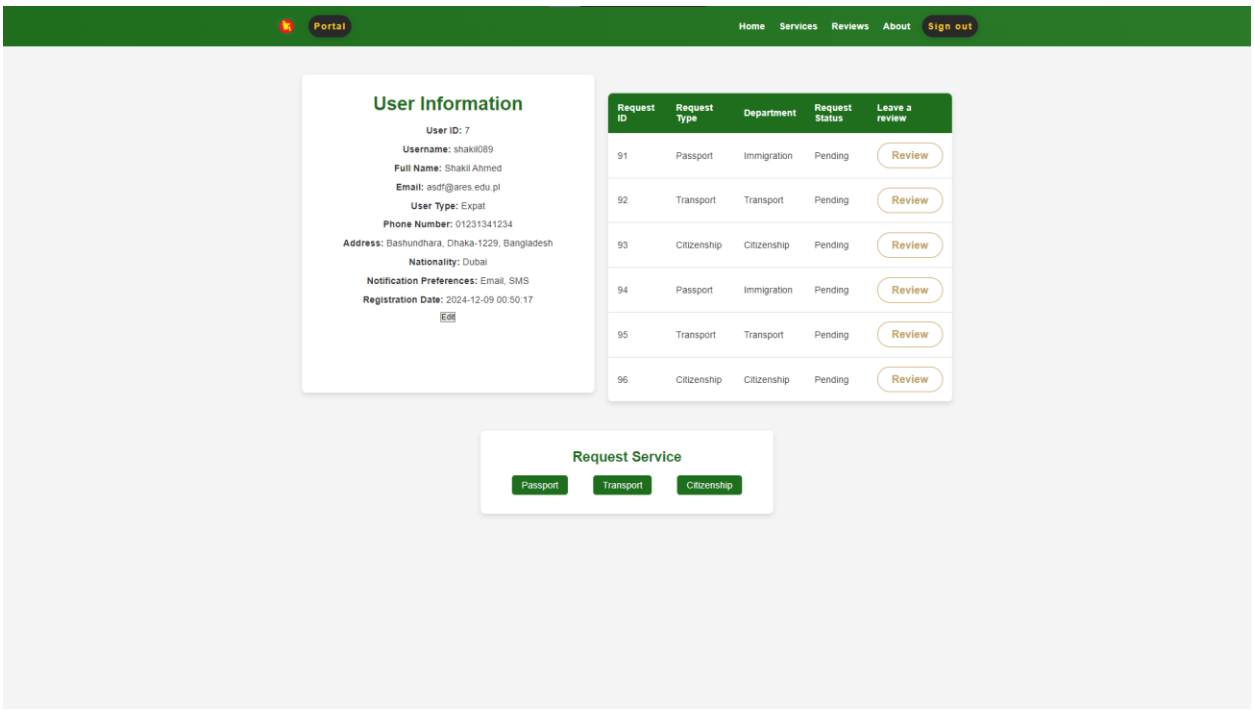
User info update Modal:



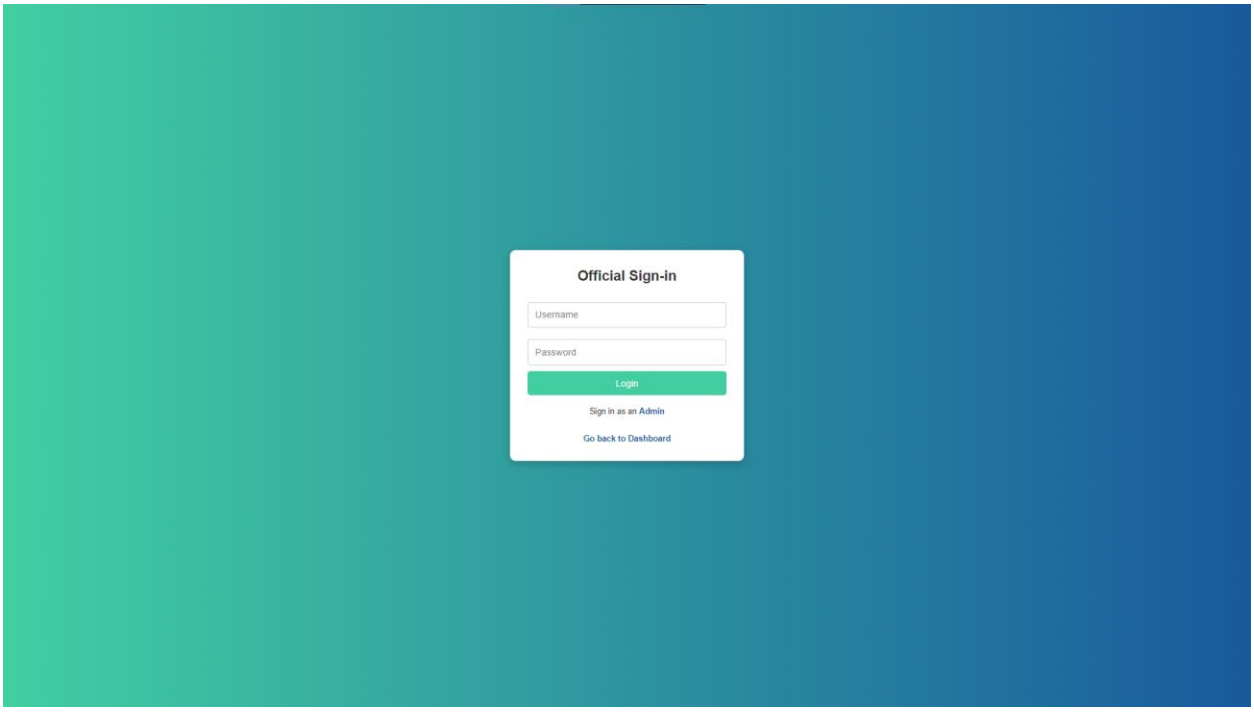
User Service Request Confirmation modal:



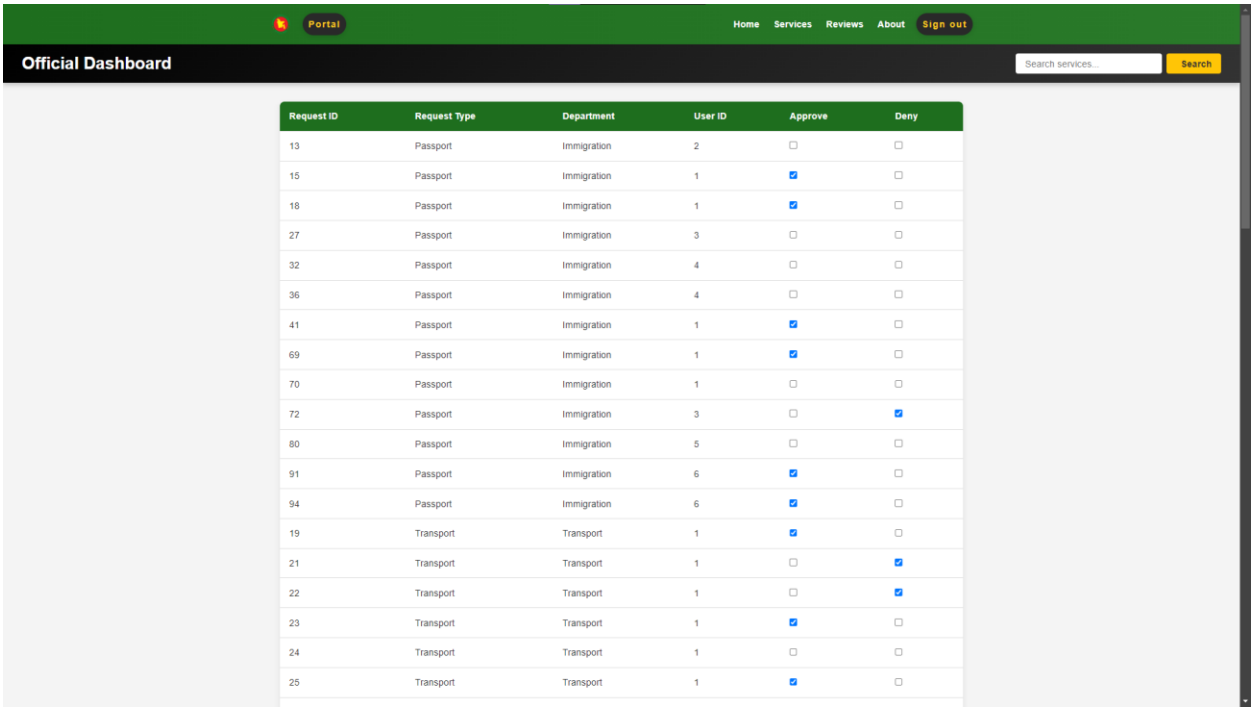
Requested services (with Request Status) shows up in User Dashboard:



Official Sign in (Account will be given from authority):



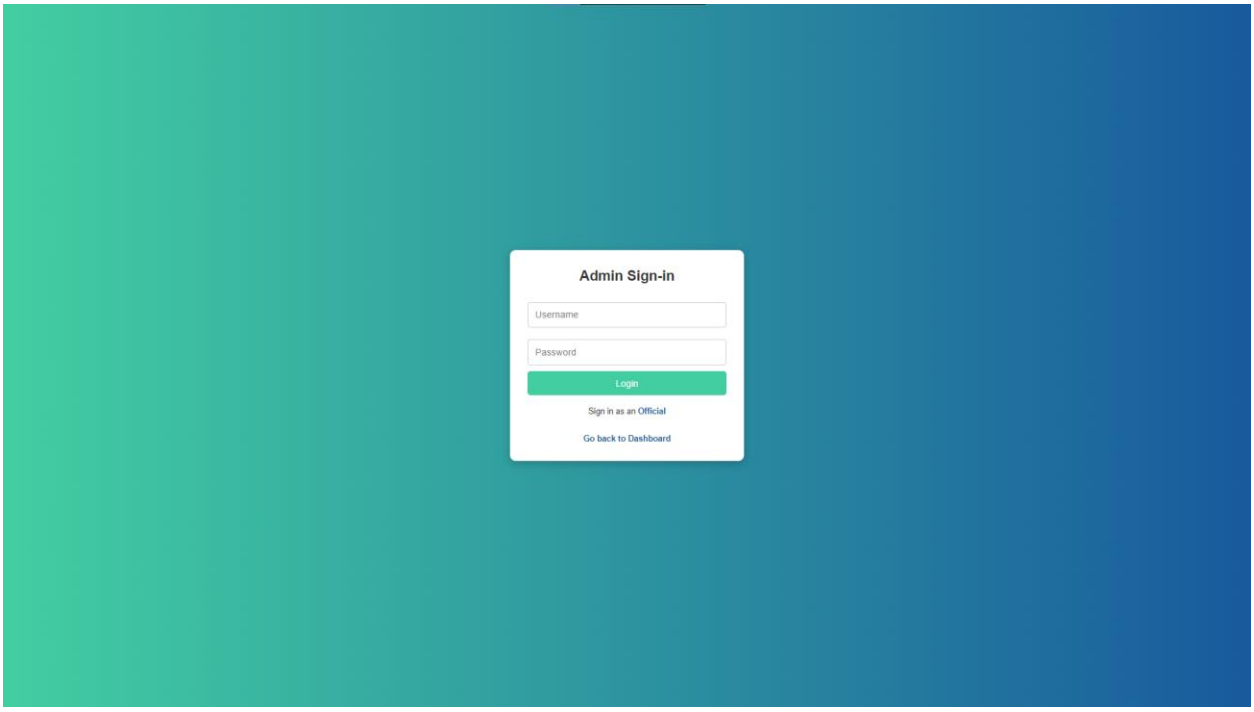
Official Dashboard(Can left at Pending, Reject or Approve and submit for Admin Approval)



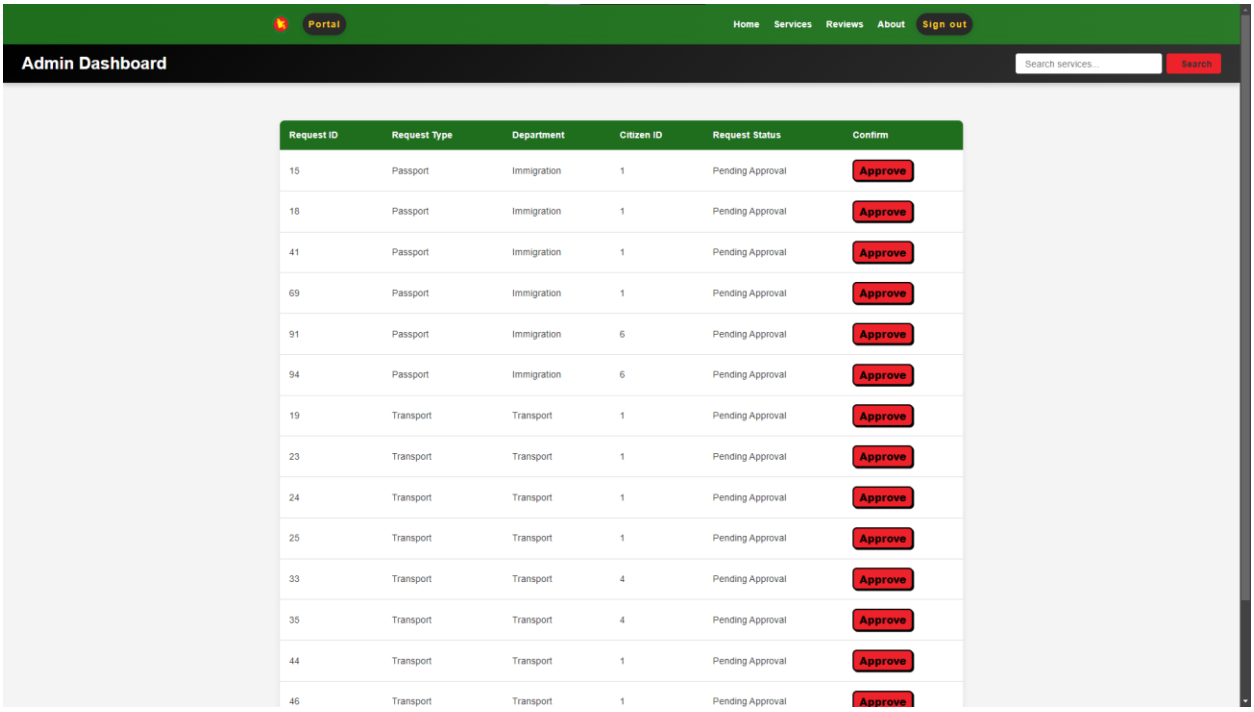
50	Citizenship	Citizenship	1	☐	☑
51	Citizenship	Citizenship	1	☐	☑
52	Citizenship	Citizenship	1	☐	☑
53	Citizenship	Citizenship	1	☐	☑
54	Citizenship	Citizenship	1	☐	☑
55	Citizenship	Citizenship	1	☐	☑
56	Citizenship	Citizenship	1	☐	☑
57	Citizenship	Citizenship	1	☐	☑
58	Citizenship	Citizenship	1	☐	☑
59	Citizenship	Citizenship	1	☐	☑
60	Citizenship	Citizenship	1	☐	☑
68	Citizenship	Citizenship	1	☐	☑
71	Citizenship	Citizenship	3	☐	☑
74	Citizenship	Citizenship	3	☐	☑
75	Citizenship	Citizenship	3	☐	☑
76	Citizenship	Citizenship	3	☐	☑
77	Citizenship	Citizenship	3	☐	☑
78	Citizenship	Citizenship	3	☐	☐
81	Citizenship	Citizenship	5	☐	☑
86	Citizenship	Citizenship	1	☑	☐
93	Citizenship	Citizenship	6	☐	☑
96	Citizenship	Citizenship	6	☑	☐

Submit

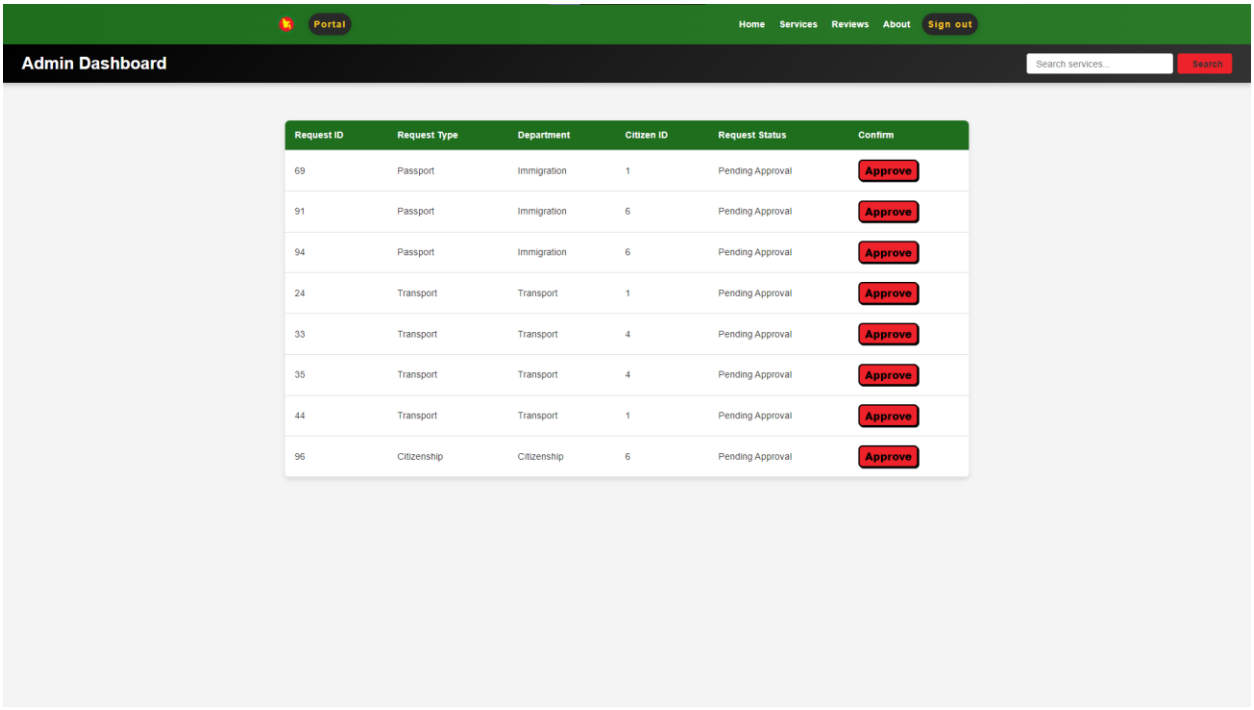
Admin Sign in (Account will be given from authority):



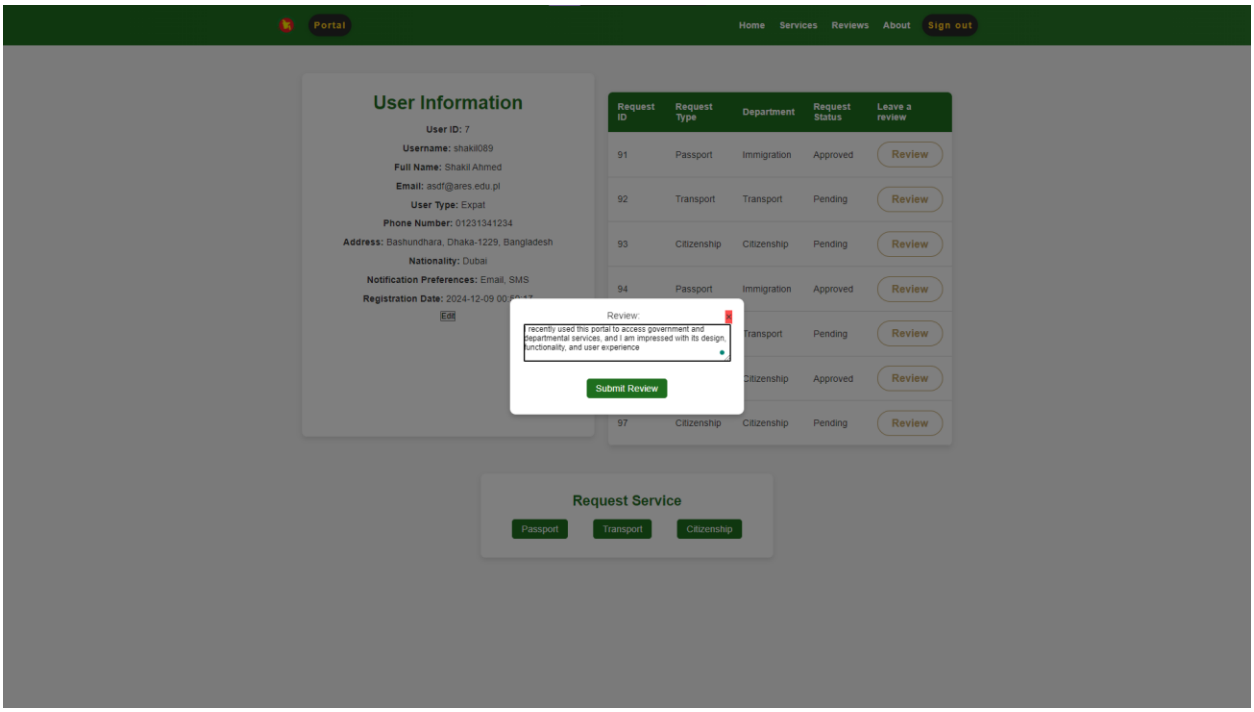
Admin Dashboard (Officials Approved requests show up for final Approval):



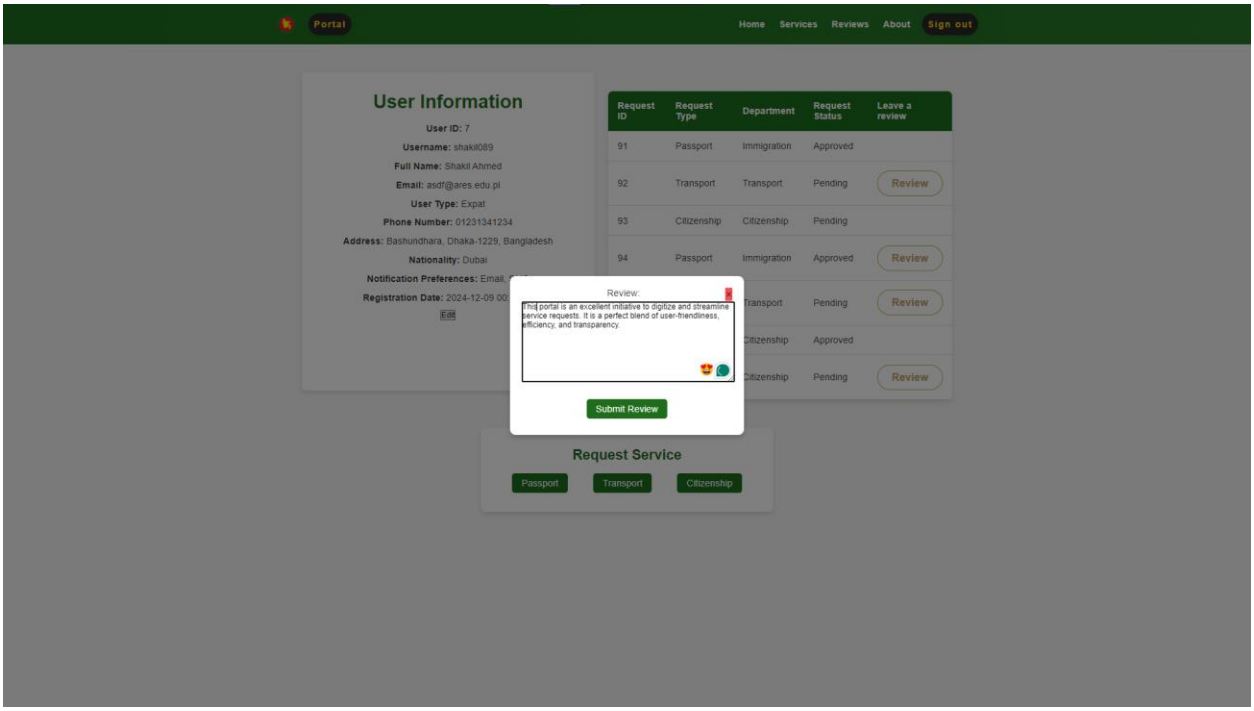
These Request are deleted upon Approval of Admin:



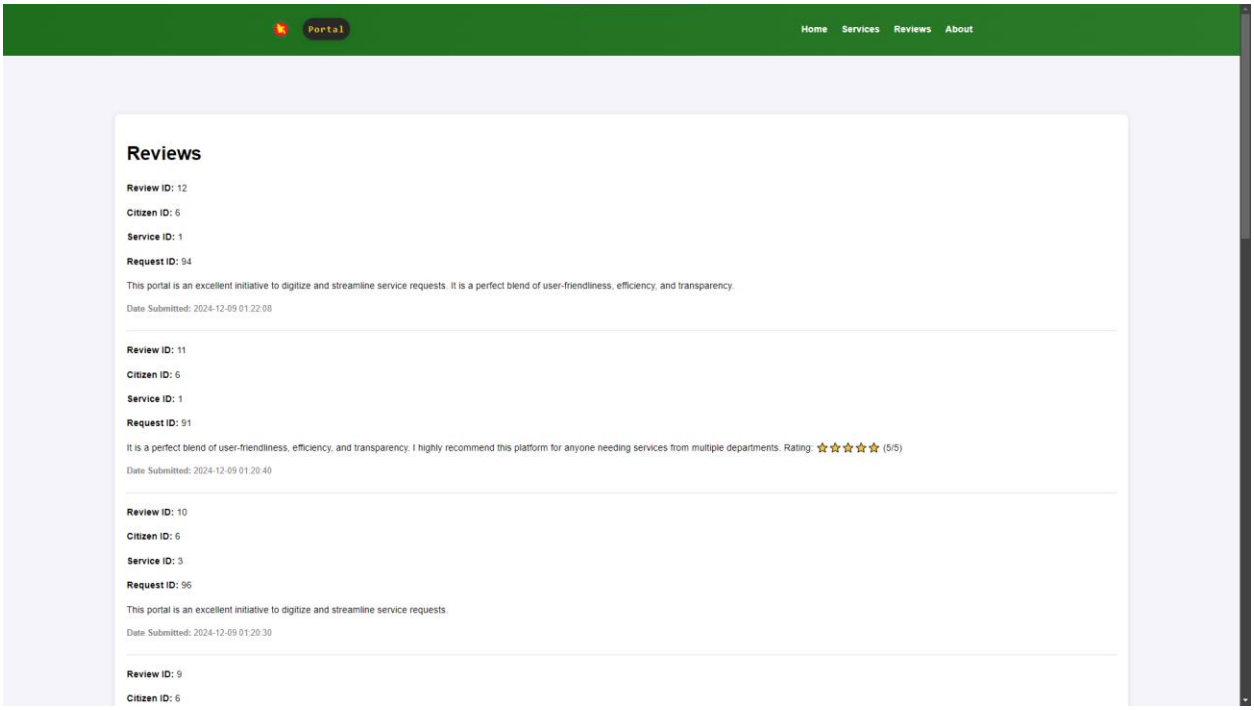
Review Section:



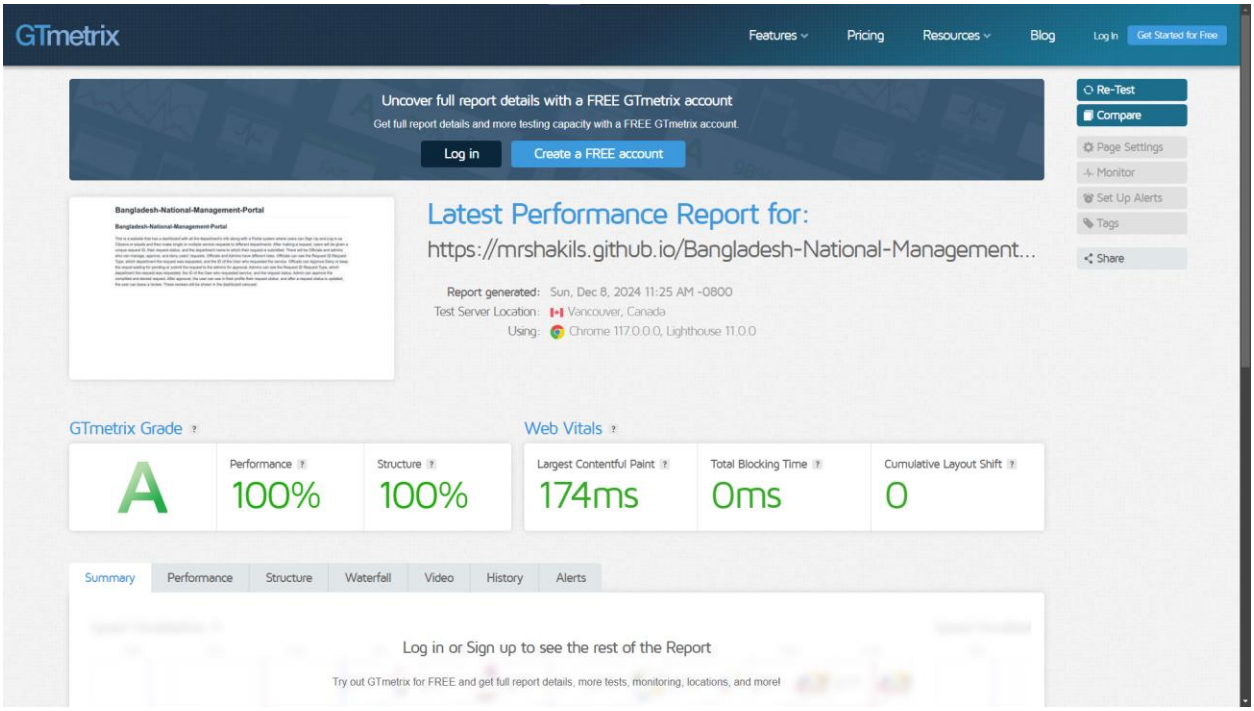
User can give review for individual Service Request:



Each Review is shown in the review page with Service info and time-date:



Website Performance Test:



Conclusion:

The **Bangladesh National Portal Management System (BNPMS)** represents a significant leap forward in integrating public service delivery into a centralized digital framework. By streamlining processes for citizens, government officials, and departments, the portal addresses critical needs for efficiency, transparency, and accessibility. Citizens benefit from enhanced engagement with government services through real-time updates, feedback mechanisms, and user-friendly interfaces. Simultaneously, government officials gain tools for better resource management and accountability, fostering trust and public satisfaction.

The system's robust database structure, comprehensive user roles, and automation capabilities ensure scalability and adaptability to future needs. As digital governance continues to evolve, BNPMS sets a precedent for harnessing technology to empower citizens and optimize administrative functions. By bridging gaps between government entities and the public, this initiative marks a pivotal step toward achieving a more efficient, connected, and transparent public service ecosystem in Bangladesh.

Part Distribution

1. *Nael*'s Contributions:

- **Entity-Relationship Diagram (ERD):**
    - **Designing the ERD for the database structure.**
- **Relation Schema:**
    - **Defining the logical schema based on the ERD.**
- **Dashboard Frontend:**
    - **Developing the user interface for the dashboard.**

2. *Shakil Ahmed*'s Contributions:

- **SQL DDL:**
    - **Writing SQL scripts for creating tables and defining relationships.**
- **Dashboard Backend:**
    - **Implementing backend functionalities for the dashboard.**
- **User, Official, and Admin Management:**
    - **Managing authentication and role-specific functionalities.**
- **Service Request System:**
    - **Developing modules for:**
        - **Request creation.**
        - **Request approval and review system.**
        - **Review section for tracking and feedback.**
- **CRUD Operations:**
    - **Ensuring all create, read, update, and delete functionalities for the portal.**
- **Project Report:**
    - **Writing and compiling the final project report.**

Contribution Ratios

Contribution Overview:

1. **Nael: 20%**
2. **Shakil Ahmed: 80%**