****

**Submitted By: group-14**

**Name: Shakil Ahmed**

**ID: 222 1453 042**

**Name: Nael**

**ID: 2212254642**

**Project: Bangladesh National Management Portal**

**Course Code: CSE311**

**Section : 12**

**Date of Submission: 02-12-2024**

**North South University**

**Department**

**of**

**Electrical and Computer Engineering**

**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**

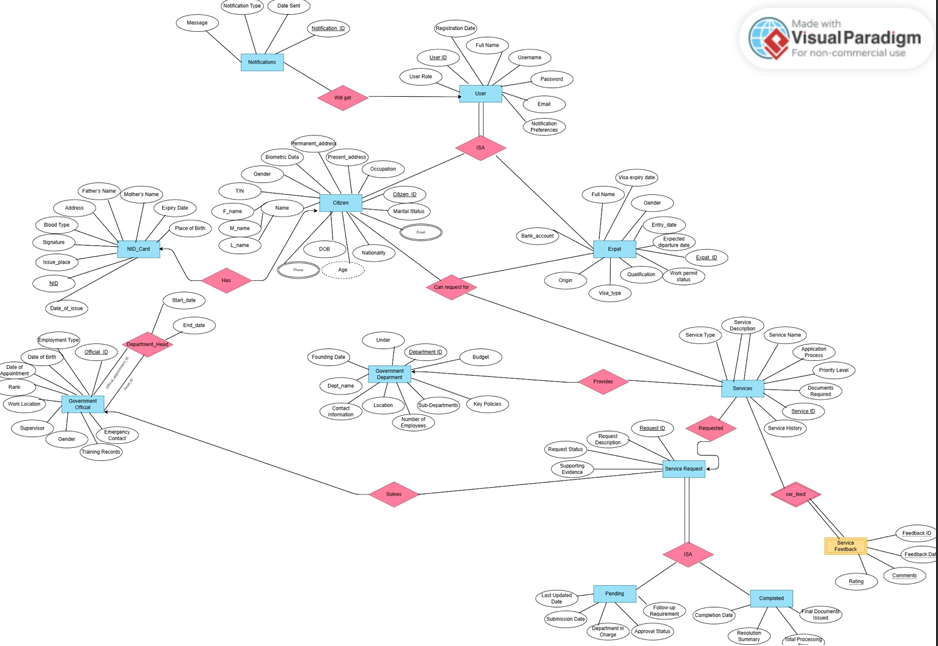
1. **User Interface and Interactions**:
   * The interface will support CRUD (Create, Read, Update, Delete) operations on major entities:
     + **Service Requests**: Citizens can submit new requests, view request statuses, update details, and delete outdated requests.
     + **Service Feedback**: Citizens 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:**

****

**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`**

**--**

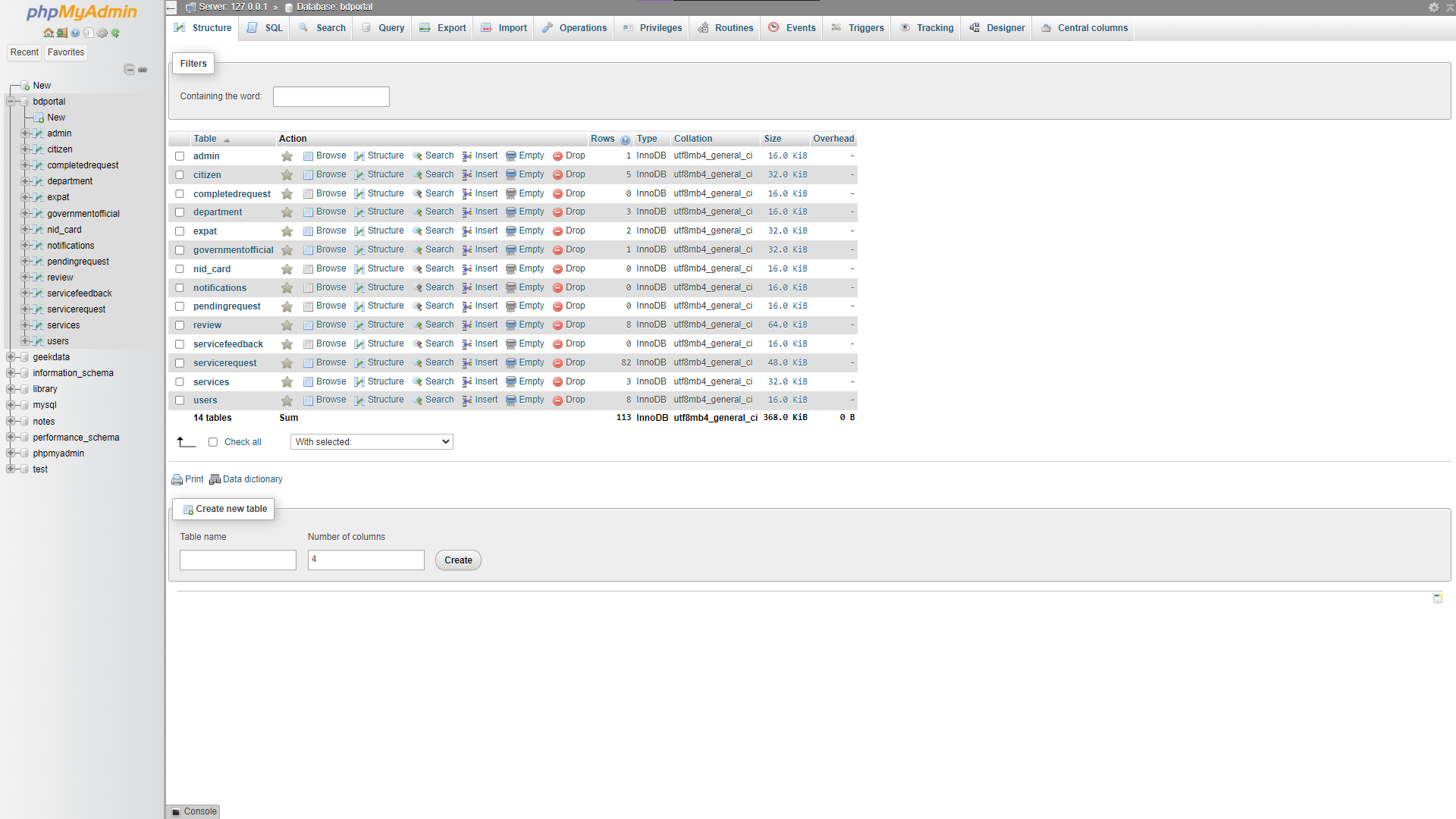
**ALTER TABLE `servicerequest`**

**ADD CONSTRAINT `servicerequest\_ibfk\_1` FOREIGN KEY (`CitizenID`) REFERENCES `citizen` (`CitizenID`),**

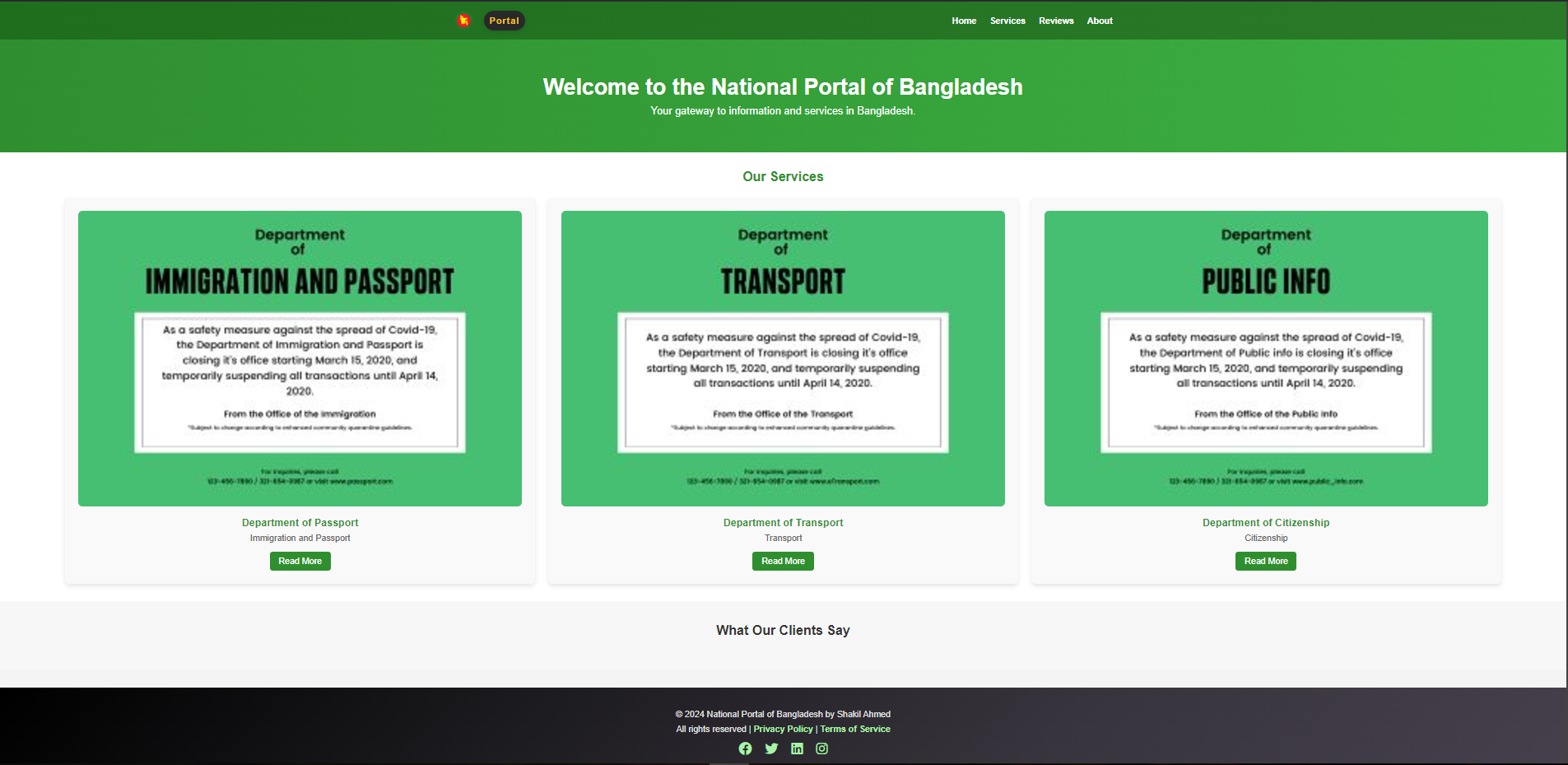
**ADD CONSTRAINT `servicerequest\_ibfk\_2` FOREIGN KEY (`ServiceID`) REFERENCES `services` (`ServiceID`);**

**COMMIT;**

**DataBase:**

****

**Dashboard** :

****

**Portal Modal offers different types of Users Portal:**

**A screenshot of a computer

Description automatically generated**

**User Sign Up (Sign Up):**

**A screen shot of a login form

Description automatically generated**

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

**A screen shot of a login form

Description automatically generated**

**User Sign in(Upon Succesful Sign up):**

**A screen shot of a login form

Description automatically generated**

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

**A screenshot of a login form

Description automatically generated**

**User DashBoard:** **A screenshot of a computer

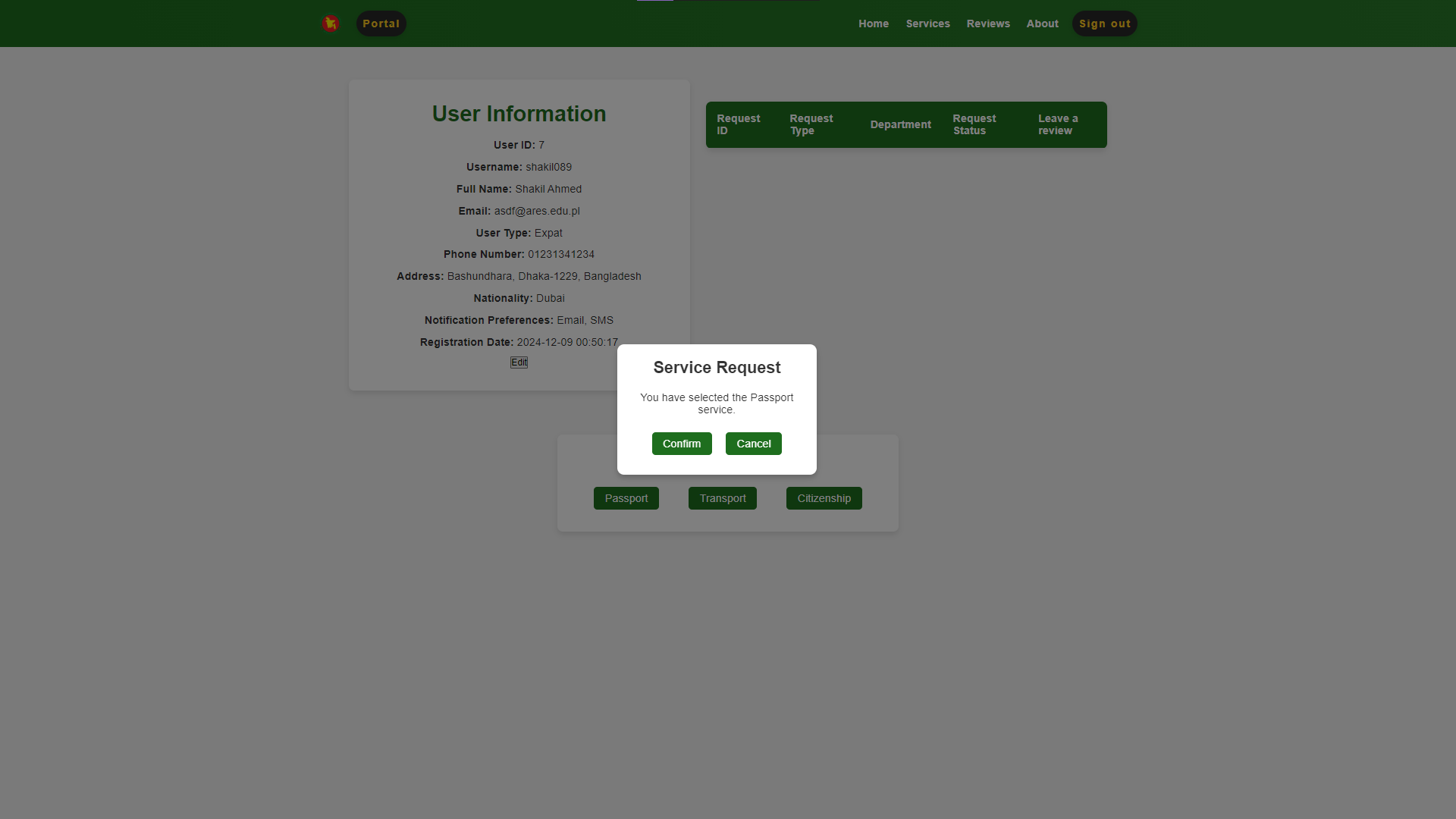
Description automatically generated**

**User info update Modal:**

**A screenshot of a computer

Description automatically generated**

**User Service Request Confirmation modal:**

****

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

**A screenshot of a computer

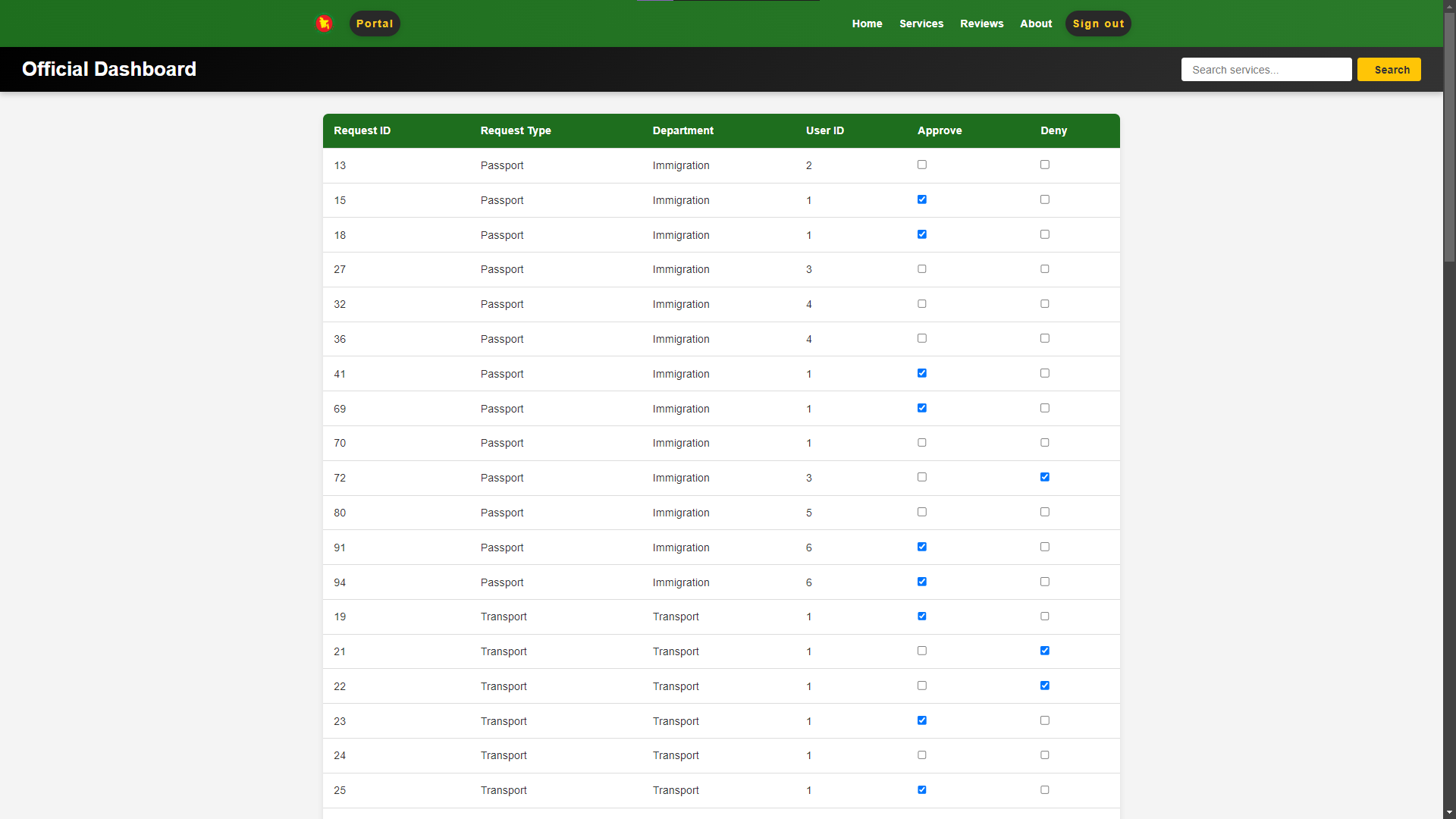
Description automatically generated**

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

**A screen shot of a login form

Description automatically generated**

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

****

**A screenshot of a computer

Description automatically generated**

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

**A screen shot of a login form

Description automatically generated**

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

**A screenshot of a computer

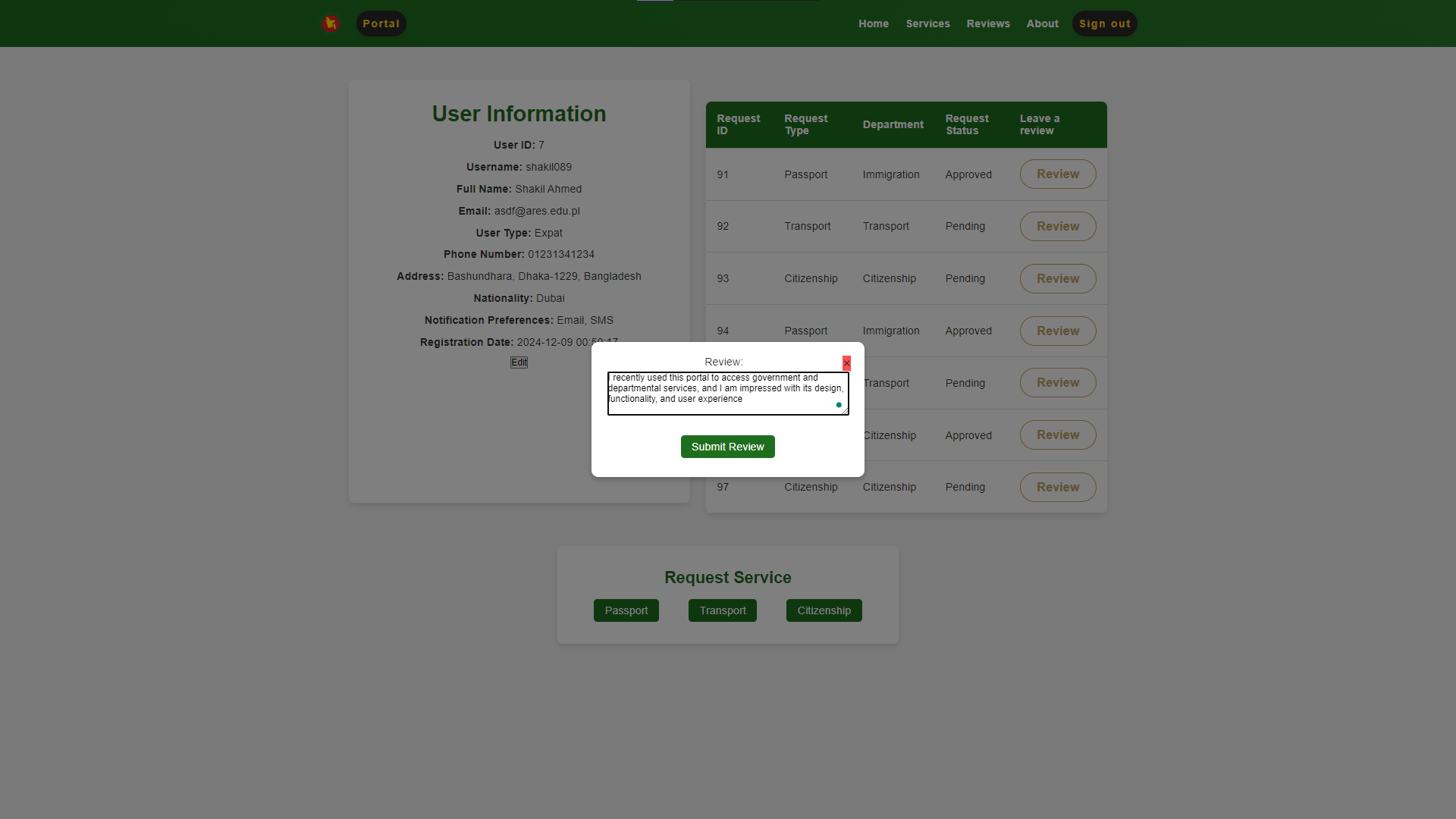
Description automatically generated**

**These Request are deleted upon Approval of Admin:**

**A screenshot of a computer

Description automatically generated**

**Review Section:**

****

**User can give review for individual Service Request:**

**A screenshot of a computer

Description automatically generated**

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

**A screenshot of a computer

Description automatically generated**

**Website Performance Test:**

**A screenshot of a computer

Description automatically generated**

**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%**

**A pie chart with a triangle in the center

Description automatically generated**