Setup Laravel & Create a New Project:

At first we have to create the project named 'payroll_management' by running following command in the CLI:

composer create-project --prefer-dist laravel/laravel payroll_management

Setting Up the Database:

After creating the MySQL database need to modify the .env file:

DB_CONNECTION=mysql

DB_HOST=127.0.0.1

DB_PORT=3306

DB_DATABASE=payroll

DB_USERNAME=root

DB_PASSWORD=

Creating Migrations & Models:

User Table: Here we are utilizing Laravel's default user migration for authentication.

Leave Categories:

php artisan make:model LeaveCategory –m

Leave Requests:

php artisan make:model LeaveRequest –m

Updating Migrations:

For leave_categories:

```php
public function up()
{
    Schema::create('leave_categories', function (Blueprint $table) {
        $table->id();
        $table->string('name');
        $table->timestamps();
    });
}
```

For leave_requests:

```php
public function up()
{
    Schema::create('leave_requests', function (Blueprint $table) {
        $table->id();
        $table->foreignId('user_id')->constrained();
        $table->foreignId('leave_category_id')->constrained();
        $table->date('start_date');
        $table->date('end_date');
        $table->string('reason');
        $table->enum('status', ['pending', 'approved', 'rejected'])->default('pending');
        $table->timestamps();
    });
}
```

User Authentication:

Use Laravel Breeze:

composer require laravel/breeze --dev

php artisan breeze:install

php artisan migrate

Employee Leave Application Management System:

Routes for web.php:

Route::resource('leave-requests', 'LeaveRequestController');

Controllers:

php artisan make:controller LeaveRequestController

Here's the implementation of the 'LeaveRequestController.php' file:

```php
<?php

namespace App\Http\Controllers;


use App\Models\LeaveRequest;

use App\Models\LeaveCategory;

use Illuminate\Http\Request;

use App\Notifications\LeaveRequestNotification;


class LeaveRequestController extends Controller

{

  public function __construct()

  {

    $this->middleware('auth');
```

```php
    }


    public function index()

    {

        $leaveRequests = LeaveRequest::where('user_id', auth()->id())->with('leave_category')->get();

        return view('leave-requests.index', compact('leaveRequests'));

    }

    public function create()

    {

        $categories = LeaveCategory::all();

        return view('leave-requests.create', compact('categories'));

    }

    public function store(Request $request)

    {

        $data = $request->validate([

            'category' => 'required|exists:leave_categories,id',

            'start_date' => 'required|date',

            'end_date' => 'required|date|after_or_equal:start_date',

            'reason' => 'required|string'

        ]);

        $leaveRequest = new LeaveRequest;

        $leaveRequest->user_id = auth()->id();

        $leaveRequest->leave_category_id = $data['category'];

        $leaveRequest->start_date = $data['start_date'];

        $leaveRequest->end_date = $data['end_date'];
```

```php
    $leaveRequest->reason = $data['reason'];

    $leaveRequest->save();

    $managers = User::role('manager')->get();

    foreach ($managers as $manager) {

        $manager->notify(new LeaveRequestNotification($leaveRequest));

    }

    return redirect()->route('leave-requests.index')->with('success', 'Leave Requested Successfully!');

}

public function show(LeaveRequest $leaveRequest)

{

    return view('leave-requests.show', compact('leaveRequest'));

}

public function edit(LeaveRequest $leaveRequest)

{

    $categories = LeaveCategory::all();

    return view('leave-requests.edit', compact('leaveRequest', 'categories'));

}

public function update(Request $request, LeaveRequest $leaveRequest)

{

    $data = $request->validate([

        'category' => 'required|exists:leave_categories,id',

        'start_date' => 'required|date',

        'end_date' => 'required|date|after_or_equal:start_date',

        'reason' => 'required|string'

    ]);
```

```php
$leaveRequest->leave_category_id = $data['category'];

$leaveRequest->start_date = $data['start_date'];

$leaveRequest->end_date = $data['end_date'];

$leaveRequest->reason = $data['reason'];

$leaveRequest->save();

return redirect()->route('leave-requests.index')->with('success', 'Leave Updated Successfully!');

    }

    public function destroy(LeaveRequest $leaveRequest)

    {

        $leaveRequest->delete();

        return redirect()->route('leave-requests.index')->with('success', 'Leave Deleted Successfully!');

    }

}
```

Blade files:

Under 'resources/views/leave-requests/user-leave-request/' need to create files index.blade.php, create.blade.php, show.blade.php and under 'resources/views/leave-requests/layouts/' need to create 'app.blade.php' file.

index.blade.php:

This will list all the leave requests. If the user is a manager, they can also approve or reject requests.

```blade
@extends('layouts.app')

@section('content')

<div class="container">

  <h1>Leave Requests</h1>

  <a href="{{ route('leave-requests.create') }}" class="btn btn-primary mb-3">Request Leave</a>
```

```
<table class="table">

    <thead>

        <tr>

            <th>ID</th>

            <th>User</th>

            <th>Leave Category</th>

            <th>Start Date</th>

            <th>End Date</th>

            <th>Reason</th>

            <th>Status</th>

            <th>Actions</th>

        </tr>

    </thead>

    <tbody>

        @foreach($leaveRequests as $request)

        <tr>

            <td>{{ $request->id }}</td>

            <td>{{ $request->user->name }}</td>

            <td>{{ $request->leave_category->name }}</td>

            <td>{{ $request->start_date }}</td>

            <td>{{ $request->end_date }}</td>

            <td>{{ $request->reason }}</td>

            <td>{{ $request->status }}</td>

            <td>

                <a href="{{ route('leave-requests.show', $request->id) }}" class="btn btn-info">View</a>
```

```blade
        @if(auth()->user()->hasRole('manager') && $request->status == 'pending')

            <!-- Here, you can add buttons to approve/reject, linking to controller methods. -->

        @endif

    </td>

  </tr>

  @endforeach

  </tbody>

  </table>

</div>

@endsection
```

create.blade.php:

This will allow users to request leave:

```blade
@extends('layouts.app')

@section('content')

<div class="container">

  <h1>Request Leave</h1>


  <form action="{{ route('leave-requests.store') }}" method="POST">

    @csrf

      <div class="mb-3">

      <label for="category" class="form-label">Leave Category</label>

      <select name="category" id="category" class="form-control">

        @foreach($categories as $category)

          <option value="{{ $category->id }}">{{ $category->name }}</option>
```

```blade
        @endforeach

      </select>

    </div>

    <div class="mb-3">

      <label for="start_date" class="form-label">Start Date</label>

      <input type="date" class="form-control" id="start_date" name="start_date" required>

    </div>

    <div class="mb-3">

      <label for="end_date" class="form-label">End Date</label>

      <input type="date" class="form-control" id="end_date" name="end_date" required>

    </div>

    <div class="mb-3">

      <label for="reason" class="form-label">Reason</label>

      <textarea class="form-control" id="reason" name="reason" rows="4" required></textarea>

    </div>

    <button type="submit" class="btn btn-primary">Submit</button>

  </form>

</div>

@endsection
```

show.blade.php:

This will display the details of a particular leave request:

```blade
@extends('layouts.app')

@section('content')

<div class="container">
```

```html
<h1>Leave Request #{{ $leaveRequest->id }}</h1>

<p><strong>User:</strong> {{ $leaveRequest->user->name }}</p>

<p><strong>Leave Category:</strong> {{ $leaveRequest->leave_category->name }}</p>

<p><strong>Start Date:</strong> {{ $leaveRequest->start_date }}</p>

<p><strong>End Date:</strong> {{ $leaveRequest->end_date }}</p>

<p><strong>Reason:</strong> {{ $leaveRequest->reason }}</p>

<p><strong>Status:</strong> {{ $leaveRequest->status }}</p>

@if(auth()->user()->hasRole('manager') && $leaveRequest->status == 'pending')

@endif
</div>
@endsection
```

User Types & Permissions:

composer require spatie/laravel-permission

Migrate the permission tables:

php artisan vendor:publish --provider="Spatie\Permission\PermissionServiceProvider" --tag="migrations"

php artisan migrate

In User model:

use Spatie\Permission\Traits\HasRoles;

class User extends Authenticatable

{

   use HasRoles;

}

Assign roles:

```php
$user->assignRole('employee');

$manager->assignRole('manager');
```

Leave Balances & Reporting:

In User model:

```php
public function leaveBalance()

{

    return 20 - $this->leave_requests()->where('status', 'approved')->count();

}
```

Sending Notifications:

First, need to create a new notification:

```
php artisan make:notification LeaveRequestNotification
```

In LeaveRequestNotification:

```php
public function toMail($notifiable)

{

    return (new MailMessage)

            ->line('You have a new leave request.')

            ->action('View Request', url('/leave-requests/'.$this->leaveRequest->id))

            ->line('Thank you for using our application!');

}
```

When a manager approves/rejects, trigger the notification:

$user->notify(new LeaveRequestNotification($leaveRequest));