

## 1.Certificate\_types

filed_name	type	rule	remarks
Id	integer	required	
certificate_name	text	required	
certificate_description	text	nullable	
is_active	Boolean	required	

## certificate\_requests

filed_name	type	rule	remarks
id	integer	required	
student_id	bigint	required	Primary key of student table
certificate_type_id	bigint	required	Primary key of certificate table
request_date	Date	required	Certificate date
purpose	text	Not required	Purpose of request
status	enum	required	ENUM('Pending', 'Approved', 'Rejected', 'Issued') DEFAULT 'Pending',
requested_by	bigint	required	Request by parent/staff/student
approved_by	bigintiger	required	Who approve the request
approved_date		required	Certificate approved date
issued_by	Big integer	required	Primary key of issued person
issued_date	date	required	Approved date
remarks	text	Not required	Add in both case request or approval

## USECASES

USECASE ID	00
Name	Seed Certificate Types
Actor(s)	System (Developer/Deployment Script)
Trigger	Application is first installed or database is reset/re-migrated
Preconditions	<p>1 Database is set up and migrations have been run.</p> <p>2 Seeder file with certificate types exists in <code>database/seeders</code> directory.</p> <p>3 The developer has execution rights for artisan commands.</p>
Main Flow	<ol style="list-style-type: none"> <li>1. Developer runs <code>php artisan db:seed --class=CertificateTypesSeeder</code>.</li> <li>2. System reads predefined certificate types from seeder file.</li> <li>3. Inserts each certificate type into <code>certificate_types</code> table with: <ul style="list-style-type: none"> <li>- Name (e.g., Bonafide, Transfer Certificate, Conduct Certificate)</li> <li>- Description</li> <li>- Template file path</li> <li>- Active status (<code>TRUE</code>).</li> </ul> </li> <li>4. Seeder confirms records were inserted successfully.</li> </ol>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> Table already has records → Seeder may skip duplicates or overwrite depending on logic.</li> <li>- <b>E2:</b> Template file path missing → Seeder logs error and skips that record.</li> <li>- <b>E3:</b> Database connection error → Seeder execution fails.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>- <code>certificate_types</code> table contains the predefined set of certificate types ready for use in requests.</li> <li>- Admin and users can see these types in dropdown lists when making requests.</li> </ul>

USECASE ID	02
Name	Request Certificate
Actor(s)	student/parent /direct office
Trigger	Student/Parent needs an official school certificate
Preconditions	<p>1. Students exist in ERP with valid status.</p> <p>2. Requested certificate type exists in <code>certificate_types</code> and is <code>Active</code></p>
Main Flow	<ol style="list-style-type: none"> <li>1. The actor logs into the ERP portal (web or mobile).</li> <li>2. Navigates to "Certificate Requests".</li> <li>3. Click "New Request".</li> <li>4. Selects certificate type from dropdown.</li> <li>5. Enter purpose/reason.</li> <li>6. Submit the request.</li> <li>7. System saves request in <code>certificate_requests</code> with <code>Pending</code> status, logs timestamp and <code>requested_by</code>.</li> </ol>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> Certificate type not active → system shows error.</li> <li>- <b>E2:</b> Student has outstanding fees (if business rule applies) → block request with reason.</li> <li>- <b>E3:</b> Network failure before saving → prompt retry.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>- Request stored in DB.</li> <li>- Notification sent to Admin.</li> <li>- Requests visible in the student portal.</li> </ul>

USECASE ID	03
Name	List and filter all certificate lists
Actor(s)	student/parent /direct office
Trigger	Admin/staff/patent/student opens the "Certificate Requests" list page(based on the permission we adjust the visibility )
Preconditions	<ul style="list-style-type: none"> <li>- Actor has permission to view requests.</li> <li>- At least one record exists in <b>certificate_requests</b>.</li> </ul>
Main Flow	<ol style="list-style-type: none"> <li>1. logs in.</li> <li>2. Open the "Certificate Requests" page.</li> <li>3. The system loads all requests.</li> <li>4. Admin applies filters:             <ul style="list-style-type: none"> <li>a) By status</li> <li>d) By date range.</li> </ul> </li> <li>5. Filtered results shown.</li> <li>6. Admin selects a request to review.</li> </ol>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> No matches → Show "No requests found".</li> <li>- <b>E2:</b> Invalid filter values → Show validation error.</li> <li>- <b>E3:</b> Large data load timeout → Show partial results.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>- Filtered list ready for further action</li> </ul>

USECASE ID	04
Name	Review the certificates
Actor(s)	student/parent /direct office
Trigger	School Admin / Designated Staff
Preconditions	Admin selects a pending request from the list
Main Flow	<ol style="list-style-type: none"> <li>1. Admin opens request.</li> <li>2. Reviews student data, request details, and eligibility.</li> <li>3. Chooses:               <ol style="list-style-type: none"> <li>a) Approve → Set <b>Approved</b>, record approver and date.</li> <li>b) Reject → Set <b>Rejected</b>, enter remarks.</li> </ol> </li> <li>4. System updates status.</li> <li>5. Notification sent to requester.</li> </ol>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> Missing student details → Cannot approve.</li> <li>- <b>E2:</b> Type retired after request → Auto reject.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>- Approved requests proceed to generation.</li> <li>- Rejected requests closed.</li> </ul>

USECASE ID	05
Name	Generate certificates
Actor(s)	School Admin/staff
Trigger	School Admin / Designated Staff
Preconditions	Request is approved and ready for document preparation
Main Flow	<ul style="list-style-type: none"> <li>- Status = <b>Approved</b>.</li> <li>- Certificate type has a valid template.</li> <li>- Student data complete.</li> </ul>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> Missing template → Error.</li> <li>- <b>E2:</b> Merge field mismatch → Highlight missing data.</li> <li>- <b>E3:</b> File save error → Retry.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>- Certificate ready for issuance.</li> </ul>

USECASE ID	06
Name	Issue Certificate
Actor(s)	School Admin / Staff
Trigger	Certificate is generated and ready for delivery
Preconditions	<ul style="list-style-type: none"> <li>- File exists in <code>certificate_attachments</code>.</li> <li>- Status = <b>Approved</b> or Ready to Issue.</li> </ul>
Main Flow	<ol style="list-style-type: none"> <li>1. Admin opens request.</li> <li>2. Verifies certificate.</li> <li>3. Marks as <b>Issued</b>.</li> <li>4. Records issuer and date.</li> <li>5. If digital → Sends link via portal/email.</li> <li>6. If physical → Prints and records collection.</li> </ol>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> File corrupted → Regenerate.</li> <li>- <b>E2:</b> Student without ID proof → Delay issuance.</li> <li>- <b>E3:</b> Email bounce → Retry delivery.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>- Certificate delivered.</li> <li>- Status updated to <b>Issued</b>.</li> </ul>

USECASE ID	07
Name	View Certificate History
Actor(s)	Student / Parent / Admin
Trigger	User wants to see past or current requests

Preconditions	<ul style="list-style-type: none"> <li>- The user has permission to view history.</li> <li>- At least one record exists for that user.</li> </ul>
Main Flow	<ol style="list-style-type: none"> <li>1. The user logs in.</li> <li>2. Opens "Certificate History".</li> <li>3. The system loads request records.</li> <li>4. Displays type, dates, status, and download link (if available).</li> <li>5. Users can click to view full details.</li> </ol>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> No records → Show “No requests yet”.</li> <li>- <b>E2:</b> File missing for issued certificate → Show “Contact admin”.</li> </ul>
Postconditions	Use can track all certificate activity

USECASE ID	08
Name	Reject Certificate Request
Actor(s)	Admin / Authorized Staff
Trigger	Admin decides that a certificate request cannot be processed due to invalid data, ineligibility, or policy violations.
Preconditions	<ul style="list-style-type: none"> <li>- Certificate request exists in <code>certificate_requests</code> table.</li> <li>- Request status is <b>Pending</b> or <b>Under Review</b>.</li> <li>- Admin has permission to reject requests.</li> </ul>

Main Flow	<ol style="list-style-type: none"> <li>1. Admin navigates to certificate request list .</li> <li>2. Opens a request in review mode .</li> <li>3. Selects "Reject" option.</li> <li>4. Enters rejection reason in a required text field.</li> <li>5. System updates request status to <b>Rejected</b>.</li> <li>6. System stores rejection reason in request record.</li> <li>7. System notifies student/parent via ERP portal or email about the rejection.</li> </ol>
Alternative Flow / Exceptions	<ul style="list-style-type: none"> <li>- <b>E1:</b> Admin tries to reject an already approved or rejected request → System shows error message.</li> <li>- <b>E2:</b> Rejection reason left blank → System prompts to enter reason.</li> <li>- <b>E3:</b> Notification service fails → Request is still rejected, but error logged for notification failure.</li> </ul>
Postconditions	<ul style="list-style-type: none"> <li>- Request status is updated to <b>Rejected</b>.</li> <li>- Reason is recorded for audit/history.</li> <li>- Student/parent sees rejection status and reason in UC-05 (View Certificate History).</li> </ul>

UI not completed