Software Design Specification

for

BrainBox

Version 1.0.0

Created at 06/01/2025

Prepared by Group 1

FPT Quy Nhon AI Campus

Member

#	Full name	Student ID	Account	Role
1	Lê Minh Vương	QE170148	VuongLM	PM, BE, FE
2	Nguyễn Thị Thúy	QE170033	ThuyNT	QA, FE
3	Đinh Quốc Chương	QE170097	ChuongDQ	BE, FE
4	Hồ Trọng Nghĩa	QE170173	NghiaHT	DS, FE
5	Lâm Mỹ Linh	QE170023	LinhLM	BA, FE
6	Lê Mạnh Hùng	QE170213	HungLM	DS, FE

*****
PM - Project Manager
BA - Business Analyst
DS - Designer
BE - Backend Developer
FE - Frontend Developer
QA - Quality Assurance

Revision History

Version	Date	A, M, D	Assignee	Change Description
0.0.1	06/01/2025	A	NghiaHT	Setup base template of document
0.1.0	11/01/2025	A	ChuongDQ	Add Database Design Schemas
0.2.0	11/01/2025	A	HungLM	Add table description Add table details description: + \`lessons\` table + \`courses\` table + \`sessions\` table + \`vouchers\` table
0.3.0	12/01/2025	A	ChuongDQ	Add Repository Link
0.4.0	12/01/2025	A	NghiaHT	Add table details description: + \`users\` table + \`notifications\` table + \`payments\` table + \`ratings\` table + \`progress\` table Add link Figma for UI/UX Design
0.5.0	17/01/2025	A	HungLM	Add Use Case for Teacher + Register, Login, Logout (Clerk)
0.6.0	17/01/2025	A	LinhLM	Add Use Case for Admin + Manage user accounts Add Use Case for Teacher + Manage personal account information
0.7.0	18/01/2025	A	HungLM	Add Use Case for Learner + View enrolled courses
0.8.0	18/01/2025	A	ThuyNT	Add Use Case for Admin + Login
0.9.0	19/01/2025	A	ThuyNT	Add Use Case for Learner + View and Edit personal account details
0.10.0	19/01/2025	A	HungLM	Add Use Case for Teacher + Create and Manage Quizzes
0.11.0	19/01/2025	A	NghiaHT	Add Use Case for Leaner: + Register, Login & Logout

Version	Date	A, M, D	Assignee	Change Description
0.12.0	21/01/2025	A	ThuyNT	Add Use Case for Teacher: + Dashboard
0.13.0	21/01/2025	A	LinhLM	Add Use Case for Admin + Approve or reject actions of the teacher of course
0.14.0	23/01/2025	A	ThuyNT	Add Use Case for Admin + View, edit and analyze reports on users, courses, and payments
0.15.0	23/01/2025	A	NghiaHT	Add Use Case for Teacher: + Create, Edit and Delete new courses with multimedia content
0.16.0	24/01/2025	A	NghiaHT	Add Use Case for Learner: + Enroll in course
0.17.0	26/01/2025	A	LinhLM	Add Use Case for Teacher: + View revenue reports for their courses
0.18.0	27/01/2025	A	NghiaHT	Add Use Case for Learner: + Track progress for each enrolled course
0.18.1	05/02/2025	M	ChuongDQ	Update Database Schemas
0.18.2	10/02/2025	M	VuongLM	Update Table Description
0.18.3	13/02/2025	M	ChuongDQ	Update Table Detail Description: - users - admins
0.18.4	14/02/2025	M	VuongLM	Update Table Detail Description: - courses - sections - lectures
0.18.5	17/02/2025	M	ChuongDQ	Update Table Detail Description: - payments - progress
0.18.6	22/02/2025	M	VuongLM	Update Table Detail Description: - conversations - messages
0.19.0	13/03/2025	A	ThuyNT	Add UC for learner: + Learners become teachers

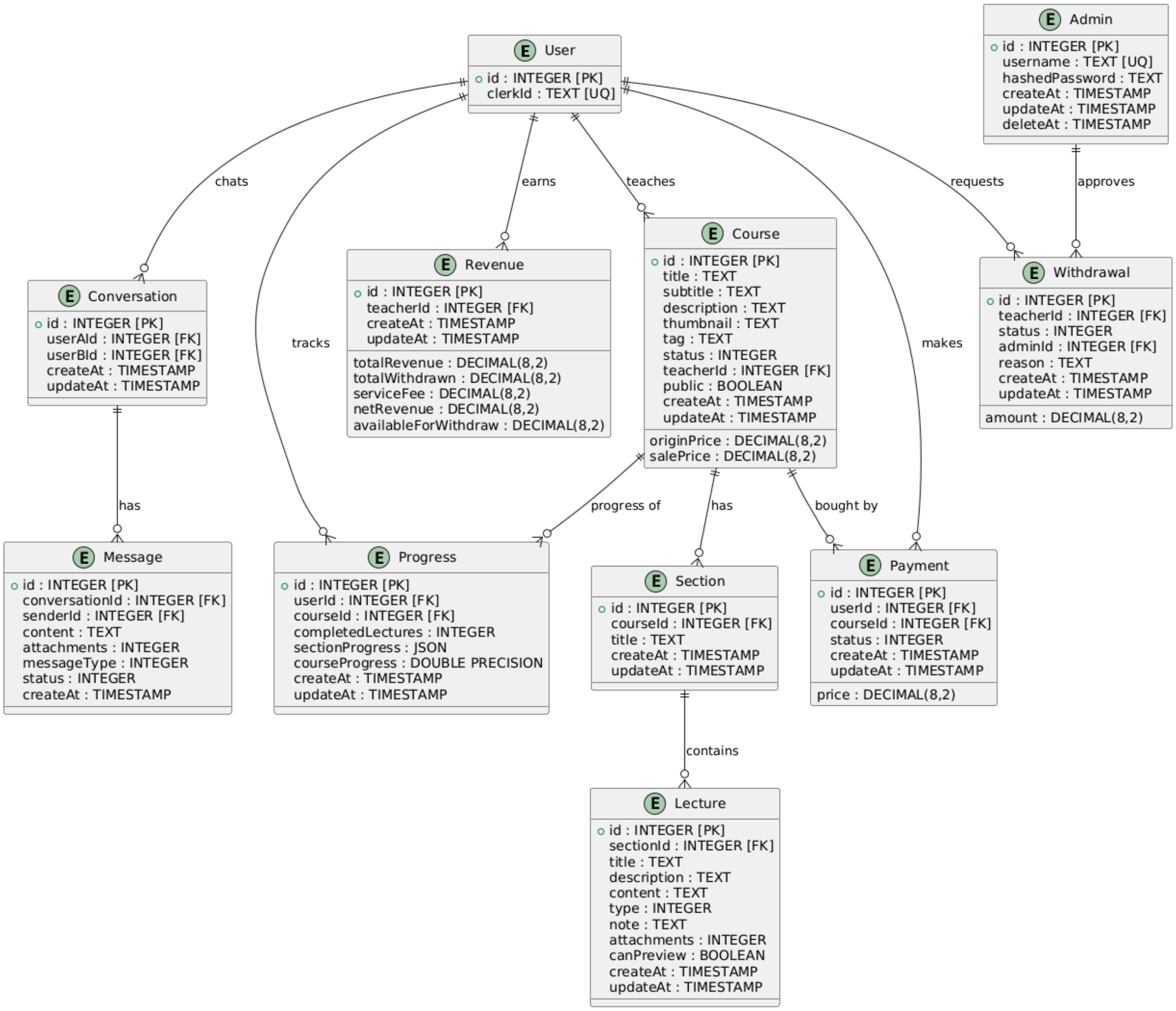
Version	Date	A, M, D	Assignee	Change Description
0.20.0	13/03/2025	A	ThuyNT	Add UC for learner: + Learner purchase history
1.0.0	26/03/2025	M	ChuongDQ	Release SDS v1.0.0

***A - Added, M - Modified, D - Deleted**

Table of contents

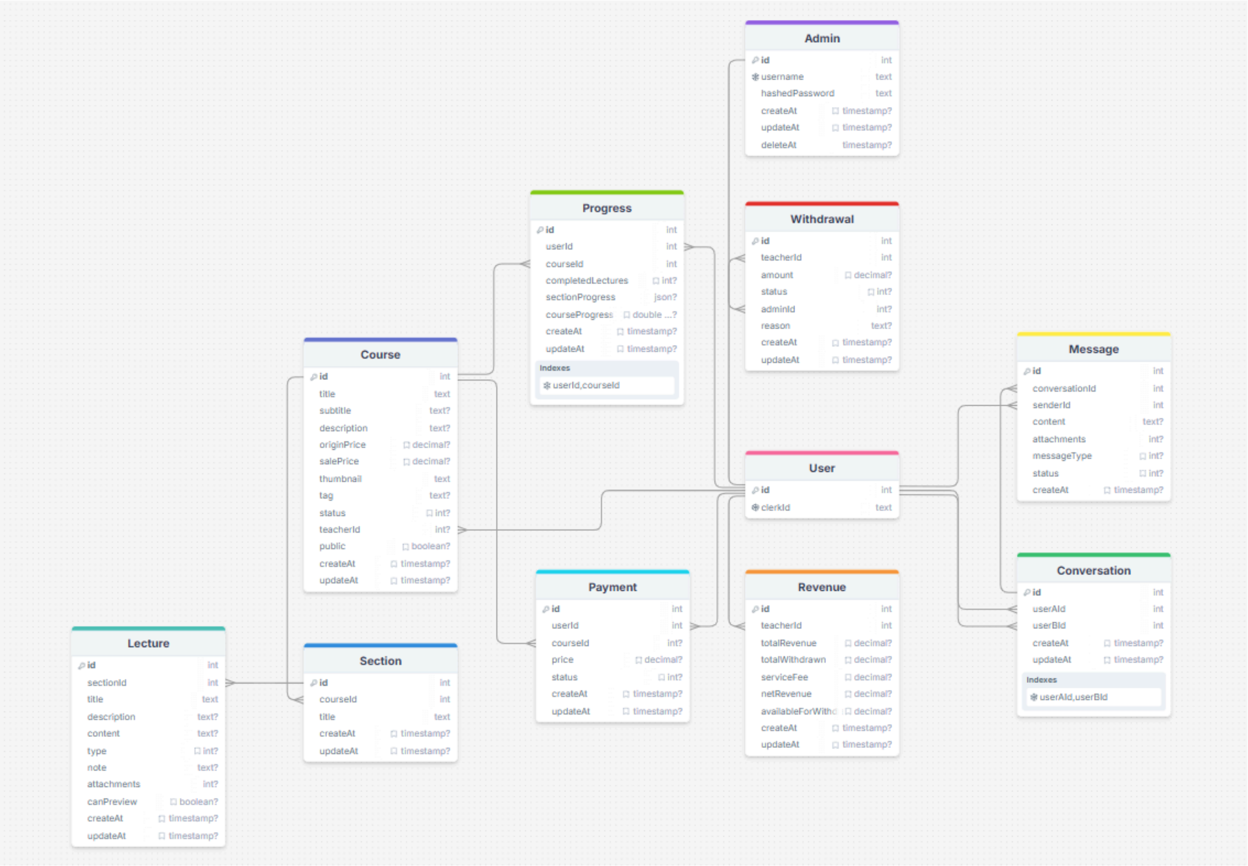
I. Overview

A. ERD Relationship



B. Database design

1. Database schemas



2. Table description

#	Table	Description
1	users	User accounts with roles (teacher, student).
2	admins	Stores admin account details.
3	courses	Stores course details, pricing, and status.
4	sections	Stores course sections and their structure.
5	lectures	Stores lesson content and resources.
6	payments	Stores payment transaction details.
7	progress	Tracks student learning progress.
8	conversations	Stores user-to-user chat conversations.
9	messages	Stores messages and attachments between users.
10	revenues	Stores financial records related to courses.

11	withdrawals	Manages teacher withdrawal requests.

3. Table details description

a. `users` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for user
2	clerkId	String	UNIQUE	Clerk authentication ID

b. `admins` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for admin
2	username	String	UNIQUE	Admin login username
3	hashedPassword	String	NOT NULL	Securely stored password
4	createAt	DateTime	DEFAULT NOW()	Record creation timestamp
5	updateAt	DateTime	AUTO_UPDATE	Last update timestamp
6	deleteAt	DateTime	NULLABLE	Soft delete timestamp

c. `courses` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for course

#	Field	Type	Constraint	Description
2	title	String	NOT NULL	Course title
3	subtitle	String	NULLABLE	Short course subtitle
4	description	String	NULLABLE	Detailed course description
5	originPrice	Decimal	DEFAULT 0.0	Original price before discount
6	salePrice	Decimal	DEFAULT 0.0	Discounted price
7	thumbnail	String	NOT NULL	Course thumbnail image URL
8	tag	String	NULLABLE	Course tag or category
9	status	ENUM	DEFAULT 'pending'	Course approval status
10	teacherId	Int	FOREIGN KEY (users.id)	Teacher who owns the course
11	public	Boolean	DEFAULT FALSE	Whether the course is public
12	createAt	DateTime	DEFAULT NOW()	Course creation timestamp
13	updateAt	DateTime	AUTO_UPDATE	Last course update timestamp

d. `sections` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for section
2	courseId	Int	FOREIGN KEY (courses.id)	Associated course ID

#	Field	Type	Constraint	Description
3	title	String	NOT NULL	Section title
4	createAt	DateTime	DEFAULT NOW()	Section creation timestamp
5	updateAt	DateTime	AUTO_UPDATE	Last section update timestamp

e. `lectures` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for lecture
2	sectionId	Int	FOREIGN KEY (sections.id)	Associated section ID
3	title	String	NOT NULL	Lecture title
4	description	String	NULLABLE	Lecture description
5	content	String	NULLABLE	Lecture content
6	type	ENUM	DEFAULT 'video'	Type of lecture content
7	note	String	NULLABLE	Additional notes
8	attachments	String[]	NULLABLE	Attached files or links
9	canPreview	Boolean	DEFAULT FALSE	Whether lecture can be previewed
10	createAt	DateTime	DEFAULT NOW()	Lecture creation timestamp
11	updateAt	DateTime	AUTO_UPDATE	Last lecture update timestamp

f. `payments` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for payment
2	userId	Int	FOREIGN KEY (users.id)	User who made the payment
3	courseId	Int	FOREIGN KEY (courses.id)	Purchased course ID
4	price	Decimal	DEFAULT 0.0	Payment amount
5	status	ENUM	DEFAULT 'pending'	Payment status
6	createAt	DateTime	DEFAULT NOW()	Payment creation timestamp
7	updateAt	DateTime	AUTO_UPDATE	Last payment update timestamp

g. `progress` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for progress
2	userId	Int	FOREIGN KEY (users.id)	User ID tracking progress
3	courseId	Int	FOREIGN KEY (courses.id)	Associated course ID
4	completedLectures	Int[]	DEFAULT []	List of completed lectures
5	sectionProgress	JSON	NULLABLE	Section-wise progress details
6	courseProgress	Float	DEFAULT 0.0	Percentage of course completed

#	Field	Type	Constraint	Description
7	createAt	DateTime	DEFAULT NOW()	Progress creation timestamp
8	updateAt	DateTime	AUTO_UPDATE	Last progress update timestamp

h. `conversations` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for conversation
2	userAId	Int	FOREIGN KEY (users.id)	First participant ID
3	userBId	Int	FOREIGN KEY (users.id)	Second participant ID
4	createAt	DateTime	DEFAULT NOW()	Conversation creation timestamp
5	updateAt	DateTime	AUTO_UPDATE	Last conversation update timestamp

i. `messages` table

#	Field	Type	Constraint	Description
1	id	Int	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for message
2	conversationId	Int	FOREIGN KEY (conversations.id)	Associated conversation ID
3	senderId	Int	FOREIGN KEY (users.id)	Message sender ID
4	content	String	NULLABLE	Message content
5	attachments	String[]	NULLABLE	Attached files or links

#	Field	Type	Constraint	Description
6	messageType	ENUM	DEFAULT 'text'	Type of message
7	status	ENUM	DEFAULT 'sent'	Message delivery status
8	createAt	DateTime	DEFAULT NOW()	Message creation timestamp

C. Repository

1. Repository

- Backend for BrainBox: https://github.com/lzaycoe/brainbox-backend
- Frontend for BrainBox: https://github.com/lzaycoe/brainbox-frontend
- Deployment for BrainBox: https://github.com/lzaycoe/brainbox-deployment
- Webhook for BrainBox: https://github.com/lzaycoe/brainbox-webhook
- Project Manager: https://app.clickup.com/9018791429/v/db/8crzng5-1158

2. CI/CD

- Backend:
    - Build:
        - Link: GitHub
        - Purpose: Build source code when push
    - Commit Message Validation:
        - Link: GitHub
        - Purpose: Validate commit messages
    - PR Validation:
        - Link: GitHub
        - Purpose: Validate PR to main branch
    - Release and Publish:
        - Link: GitHub
        - Purpose: Create release on GitHub; build & push Docker image to registry
- Frontend:
    - Commit Message Validation:
        - Link: GitHub
        - Purpose: Validate commit messages
    - Lint:
        - Link: GitHub
        - Purpose: Lint source code

- PR Validation:
  - Link: GitHub
  - Purpose: Validate PR to main branch
- Release:
  - Link: GitHub
  - Purpose: Create release on GitHub

3. Dependencies

a. Frontend

#	Package	Version	Description
1	@clerk/nextjs	6.11.1	Authentication and user management for Next.js
2	@hookform/resolvers	3.10.0	Schema validation for react-hook-form
3	@radix-ui/react-accordion	1.2.3	Customizable accordion for React
4	@radix-ui/react-avatar	1.1.3	Simple avatar display
5	@radix-ui/react-checkbox	1.1.4	Customizable checkbox for React
6	@radix-ui/react-dialog	1.1.6	Accessible dialog component
7	@radix-ui/react-dropdown-menu	2.1.6	Customizable dropdown menu
8	@radix-ui/react-label	2.1.1	Accessible label for forms
9	@radix-ui/react-progress	1.1.2	Customizable progress bar
10	@radix-ui/react-radio-group	1.2.3	Accessible radio group
11	@radix-ui/react-select	2.1.6	Customizable select component
12	@radix-ui/react-separator	1.1.2	Customizable separator line
13	@radix-ui/react-slot	1.1.2	Slot component for Radix UI
14	@radix-ui/react-toast	1.2.6	Customizable toast notifications
15	@radix-ui/react-toggle	1.1.2	Simple toggle button
16	@radix-ui/react-tooltip	1.1.8	Accessible tooltip component
17	@supabase/ssr	0.5.2	Supabase support for Next.js SSR

#	Package	Version	Description
18	axios	1.7.9	HTTP client for browsers and Node.js
19	class-variance-authority	0.7.1	Conditional CSS class utility
20	clsx	2.1.1	Conditional class name utility
21	lodash	4.17.21	JavaScript utility library
22	lucide-react	0.475.0	Modern icon set for React
23	next	15.1.6	Full-stack React framework
24	next-themes	0.4.4	Theme management for Next.js
25	react	19.0.0	Popular UI library
26	react-dom	19.0.0	React API for browser rendering
27	react-hook-form	7.54.2	High-performance form management
28	react-icons	5.4.0	Icon library for React
29	recharts	2.15.1	Charting library for React
30	tailwind-merge	2.6.0	Optimizes Tailwind CSS class merging
31	tailwindcss-animate	1.0.7	Animations for Tailwind CSS
32	zod	3.24.1	TypeScript schema validation

b. Backend

#	Package	Version	Description
1	@clerk/backend	1.24.0	Authentication backend for Clerk
2	@nestjs/common	11.0.9	Common utilities for NestJS
3	@nestjs/config	4.0.0	Configuration module for NestJS
4	@nestjs/core	11.0.9	Core module of NestJS
5	@nestjs/jwt	11.0.0	JWT authentication for NestJS
6	@nestjs/passport	11.0.5	Passport integration for NestJS
7	@nestjs/platform-express	11.0.9	Express adapter for NestJS

#	Package	Version	Description
8	@nestjs/platform-socket.io	11.0.11	Socket.IO adapter for NestJS
9	@nestjs/swagger	11.0.3	OpenAPI documentation for NestJS
10	@nestjs/websockets	11.0.11	WebSocket module for NestJS
11	@prisma/client	6.3.1	Prisma ORM client for database access
12	@types/socket.io	3.0.2	TypeScript definitions for Socket.IO
13	argon2	0.41.1	Password hashing with Argon2
14	class-transformer	0.5.1	Object transformation utilities
15	class-validator	0.14.1	Data validation decorators for TypeScript
16	cookie-parser	1.4.7	Middleware to parse cookies in Express
17	express	4.21.2	Minimalist web framework for Node.js
18	morgan	1.10.0	HTTP request logger middleware
19	nodemailer	6.10.0	Email sending library for Node.js
20	passport	0.7.0	Authentication middleware for Node.js
21	passport-custom	1.1.1	Custom authentication strategy for Passport
22	passport-jwt	4.0.1	JWT strategy for Passport
23	passport-local	1.0.0	Local authentication strategy for Passport
24	reflect-metadata	0.2.2	Metadata reflection for TypeScript decorators
25	rxjs	7.8.1	Reactive programming library
26	socket.io	4.8.1	Real-time WebSockets for Node.js
27	svix	1.60.0	Webhook management for applications

4. Deployment

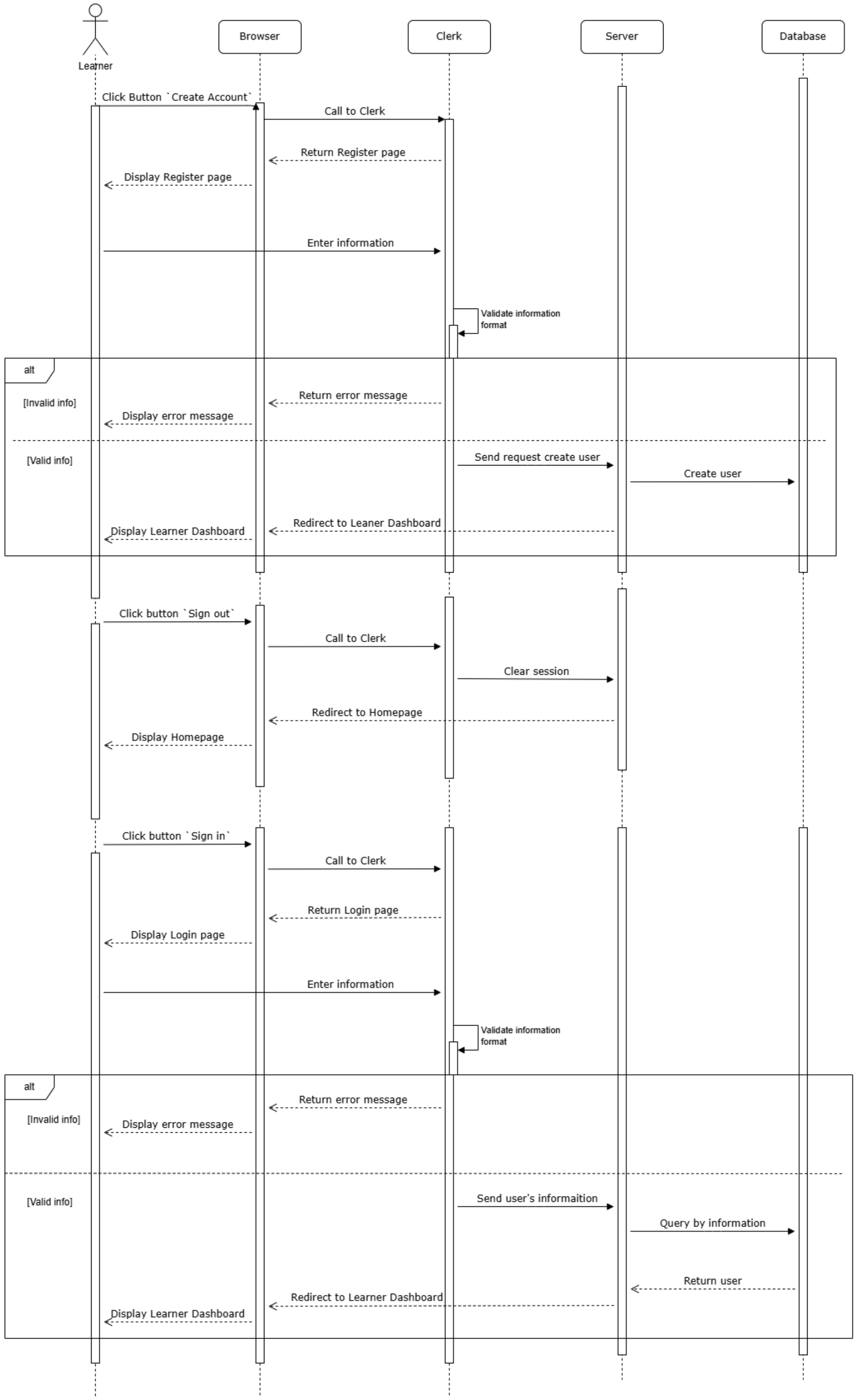
- Backend: Render
- Frontend: Vercel

II. Use Case Design

A. Learner

1. Register, Login & Logout

a) Sequence Diagram



UC-L-1 : Register, Login & Logout for Learner

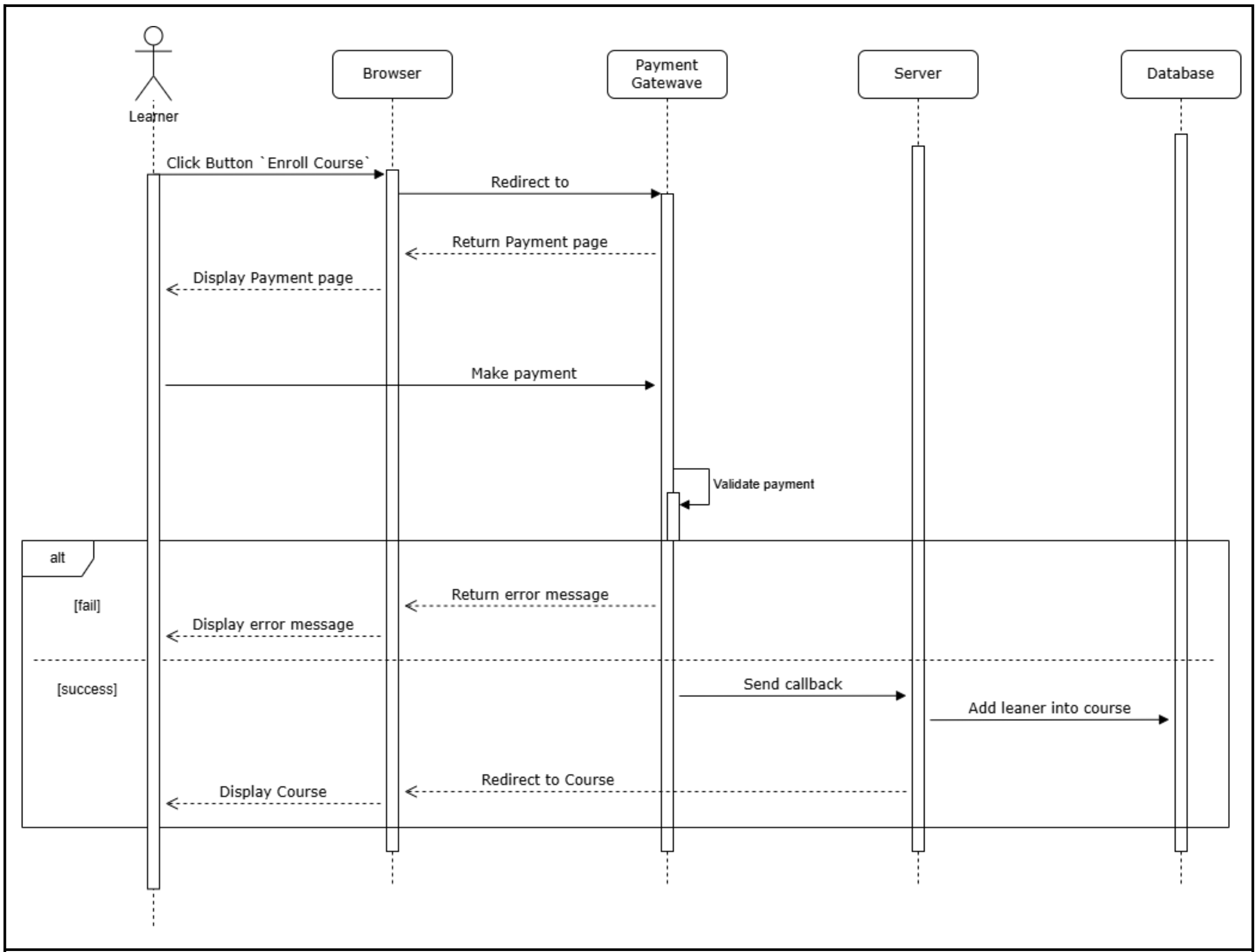
b) Flow Diagram

ID and Name	UC-L-1 : Register, Login & Logout for Learner		
Date Created	17/01/2025		
Primary Actor	Learner	Secondary Actor	CLerk
Description	A function that allows users to register as learners, then they can login to the system and log out.		
Trigger	- Register: A learner wants to register and click the button `Create Account`. - Login: A learner wants to login to the website and click the button `Sign In`. - Logout: A learner wants to logout and click the button `Sign out`.		
Preconditions	PRE-1. Learner can access the website		
Postconditions	POST-1. The BrainBox system will redirect learners to Learner's Dashboard screen when login and register success. POST-2. The BrainBox system will reload the login and register page, also display error notification when login and register fail. POST-3. The BrainBox system will redirect learners to Homepage when they logout.		
Normal Flow	**NF-1. Register** 1. Click the `Create Account` button. 2. Enter information (email, username, password,...). 3. Click the button `Continue`. **NF-2. Login** 1. Click the `Sign In` button. 2. Enter information (email, password). 3. Click the button `Continue`. **NF-3. Logout** 1. Click the user profile. 2. Click the button `Sign out`. 3. Auto access to Homepage		
Alternative Flow	**AF-1. Register with Google, Facebook, LinkedIn account** 1. Refer to step 1 of NF-1. 2. Click the buttons `Google`, `Facebook`, `LinkedIn`. 3. Refer to step 3 in NF-1. **AF-2. Login with Google, Facebook, LinkedIn account** 1. Refer to step 1 of NF-2.		

	2. Click the buttons `Google`, `Facebook`, `LinkedIn`. 3. Refer to step 3 in NF-1.
Exceptions	**E-1. Database connection failure** 1. The BrainBox System generates an exception indicating the database connection failure. 2. The Learner is notified of the issue and prompted to check network connectivity or contact technical support. **E-2. Wrong email input** 1. The BrainBox System will display an error message on the form. **E-3. Not input email** 1. The BrainBox System will display a missing field input message on the form. **E-4. Wrong password input** 1. The BrainBox System will display an error message on the form. **E-5. Not input password.** 1. The BrainBox System will display a missing field input message on the form.
Priority	High
Frequency of Use	Approximately 100 learners
Business Rules	BR-001, BR-003, BR-005 from SRS document

2. Enroll in courses (after successful payment).

a) Sequence Diagram



UC-L-2 : Enroll in courses.

b) Flow Diagram

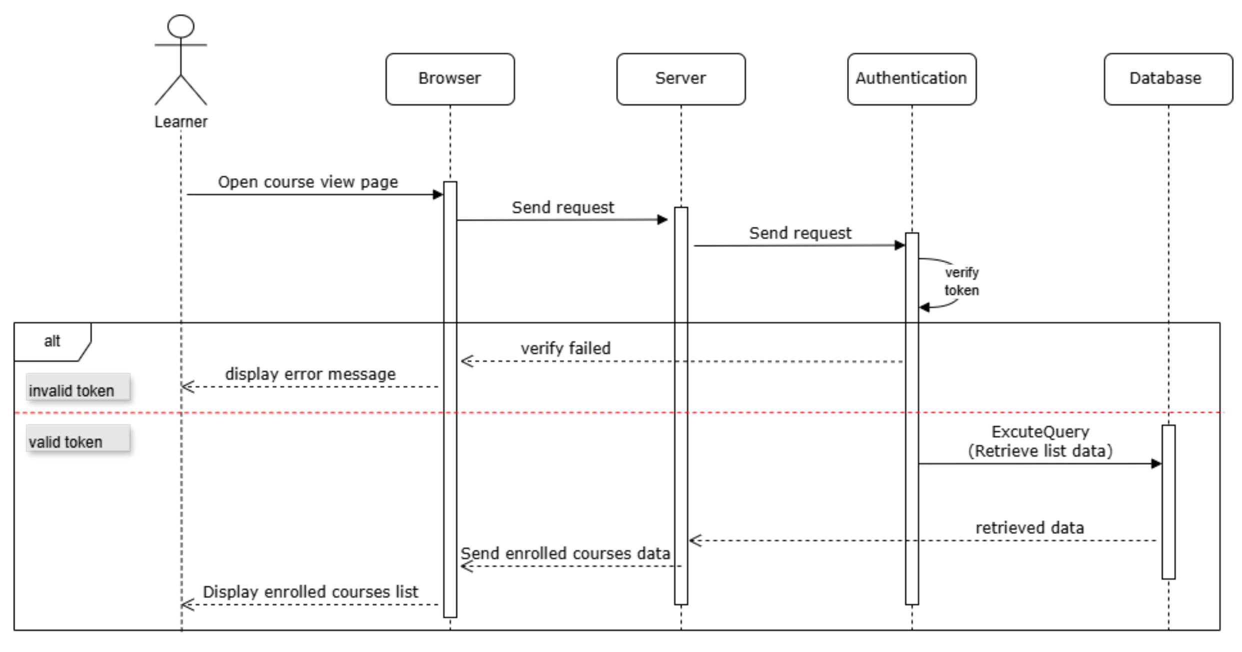
ID and Name	UC-L-2 : Enroll in courses.		
Date Created	24/01/2025		
Primary Actor	Leaner	Secondary Actor	Payment Gateway
Description	A function that allows learners to enroll into courses.		
Trigger	A learner clicks the button Enroll course through a web browser.		
Preconditions	**PRE-1.** The learner must be logged into their account.		

	PRE-2. The selected course must be published and available for enrollment. **PRE-3.** The learner must have a valid payment method linked (e.g., credit card, e-wallet). **PRE-4.** The payment gateway must be operational.
Postconditions	**POST-1**. The learner is successfully enrolled in the course, and the course is added to their "My Courses" section. **POST-2**. A payment receipt is generated and sent to the learner's registered email. **POST-3**. The enrollment record is stored in the system database. **POST-4**. The system updates the course's enrollment count.
Normal Flow	**NF-1**. **Enroll to course** 1. The learner navigates to the course details page. 2. The learner clicks the **"Enroll Course"** button. 3. The system redirects the learner to the payment gateway. 4. The learner provides payment details and confirms the payment. 5. The payment gateway processes the payment and notifies the system of the transaction status. 6. If the payment is successful: a. The system enrolls the learner in the course. b. The system generates a receipt and sends a confirmation email to the learner. c. The course is added to the learner's "My Courses" section. 7. The learner is redirected to the course overview page with a success message.
Alternative Flow	**AF-1**. **If the learner clicks "Cancel" during the payment process:** 1. The system cancels the enrollment process and redirects the learner to the course details page. **AF-2**. **If the payment gateway experiences a delay:** 1. The system displays a message: "Payment processing. Please wait..." 2. The learner is notified via email once the payment status is confirmed.

Exceptions	**E-1. The payment gateway returns a failed transaction status:** 1. The system displays an error message: "Payment failed. Please try again or use a different payment method." **E-2. The selected course becomes unavailable during the enrollment process:** 1. The system cancels the enrollment and displays a message: "This course is no longer available." **E-3. Network issues during payment processing:** 1. The system displays a message: "Network error. Please check your connection and try again." **E-4. The learner’s account is suspended or inactive:** 1. The system blocks the enrollment and displays a message: "Your account is not active. Please contact support."
Priority	High
Frequency of Use	Daily, depending on the number of learners accessing the system.
Business Rules	N/A

3. View enrolled courses.

a) Sequence Diagram



UC-L-3 : View and Edit personal account details

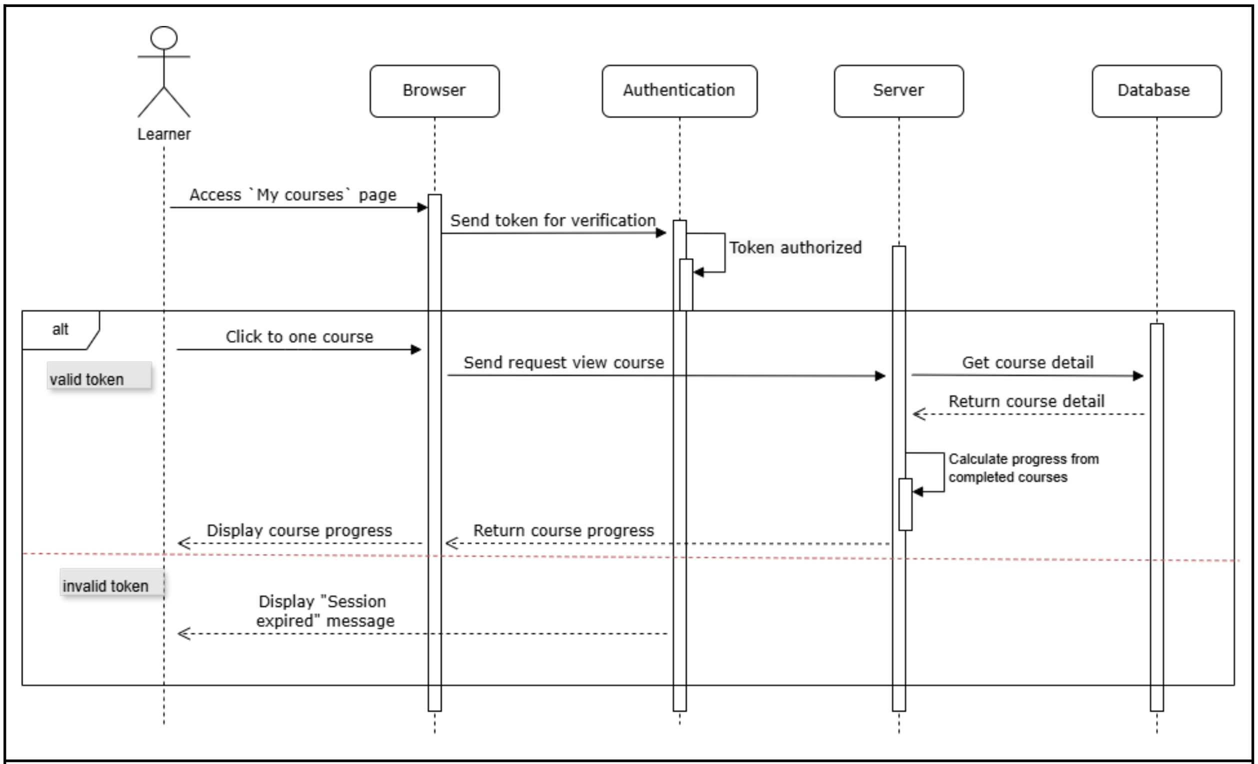
b) Flow Diagram

ID and Name	UC-L-3: View enrolled courses.		
Date Created	17/01/2025		
Primary Actor	Learner	N/A	N/A
Description	Learners view a list of courses they are currently enrolled in through their account on the platform. The system fetches the list of enrolled courses based on the learner's active session and displays it.		
Trigger	The learner accesses their course dashboard or selects an option to view their enrolled courses.		
Preconditions	PRE-1. The learner must be logged in to their account with a valid session. PRE-1. The learner must have enrolled in at least one course.		
Postconditions	POST-1. The learner views a list of courses they are enrolled in, including course details. POST-1. If the session is invalid or expired, the learner will see an error message prompting them to log in again.		

Normal Flow	1. Learner logs in successfully and accesses the course dashboard. 2. The browser sends a request to the authentication service to verify the session. 3. If the session is valid, the server fetches the list of courses the learner is enrolled in from the database. 4. The server sends the course list to the browser. 5. The browser displays the list of enrolled courses to the learner.
Alternative Flow	**Invalid Session**: If the session is invalid or expired, the server returns an authentication error, prompting the browser to show an error message, asking the learner to log in again.
Exceptions	1. If the learner has not enrolled in any course, an empty list is returned, and a message indicating no courses are available is shown. 2. If there's an error in querying the database, an error message is displayed to the learner.
Priority	High
Frequency of Use	High
Business Rules	BR-002, BR-004, BR-005, BR010 from SRS document

4. Track progress for each enrolled course.

a) Sequence Diagram



UC-L-4 : Track progress for each enrolled course.

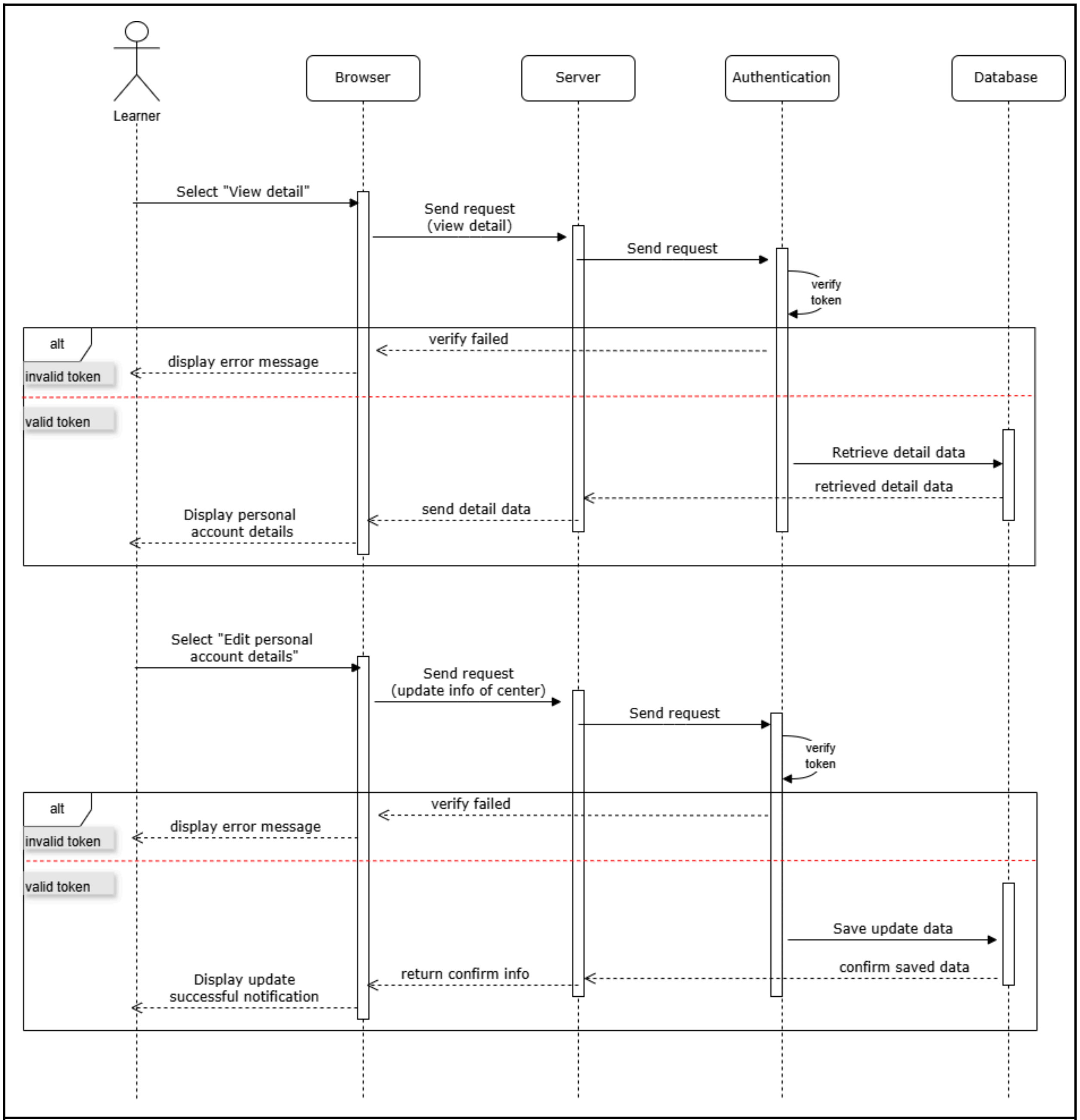
b) Flow Diagram

ID and Name	UC-L-4 : Track progress for each enrolled course.		
Date Created	27/01/2025		
Primary Actor	Learner	Secondary Actor	N/A
Description	A function that allows learners to view progress of completion for each enrolled course.		
Trigger	A learner accesses one of the courses that they have learned.		
Preconditions	**PRE-1**. The learner must be logged into their account. **PRE-2**. The learner must have at least one enrolled course. **PRE-3**. The course must contain trackable content (e.g., lessons, quizzes, assignments).		

Postconditions	**POST-1**. The learner can view the progress percentage of the enrolled course. **POST-2**. The system updates the progress in real-time as the learner completes lessons or activities. **POST-3**. The learner can view details of completed and remaining tasks within the course.
Normal Flow	**NF-1**. **Track the progress** 1. The learner logs into their account and navigates to the "My Courses" section. 2. The learner selects an enrolled course to view its progress. 3. The system retrieves the learner's progress data for the selected course. 4. The system displays the following details: a. Overall progress as a percentage. b. List of completed lessons and activities. c. List of pending lessons and activities. d. Estimated time to complete the course (if applicable). 5. The learner reviews their progress and resumes the course from where they left off.
Alternative Flow	**AF-1. If the learner has no enrolled courses:** 1. The system displays a message: "You have not enrolled in any courses yet." 2. The learner is prompted to browse available courses.
Exceptions	**E-1. The system fails to retrieve progress data due to server issues:** 1. The system displays an error message: "Unable to fetch progress. Please try again later." **E-2. The learner's session expires during the operation:** 2. The system redirects the learner to the login page.
Priority	High
Frequency of Use	Daily, depending on how often learners access their courses.
Business Rules	BR-002 from SRS document

5. View and Edit personal account details.

a) Sequence Diagram



UC-L-5 : View and Edit personal account details

b) Flow Diagram

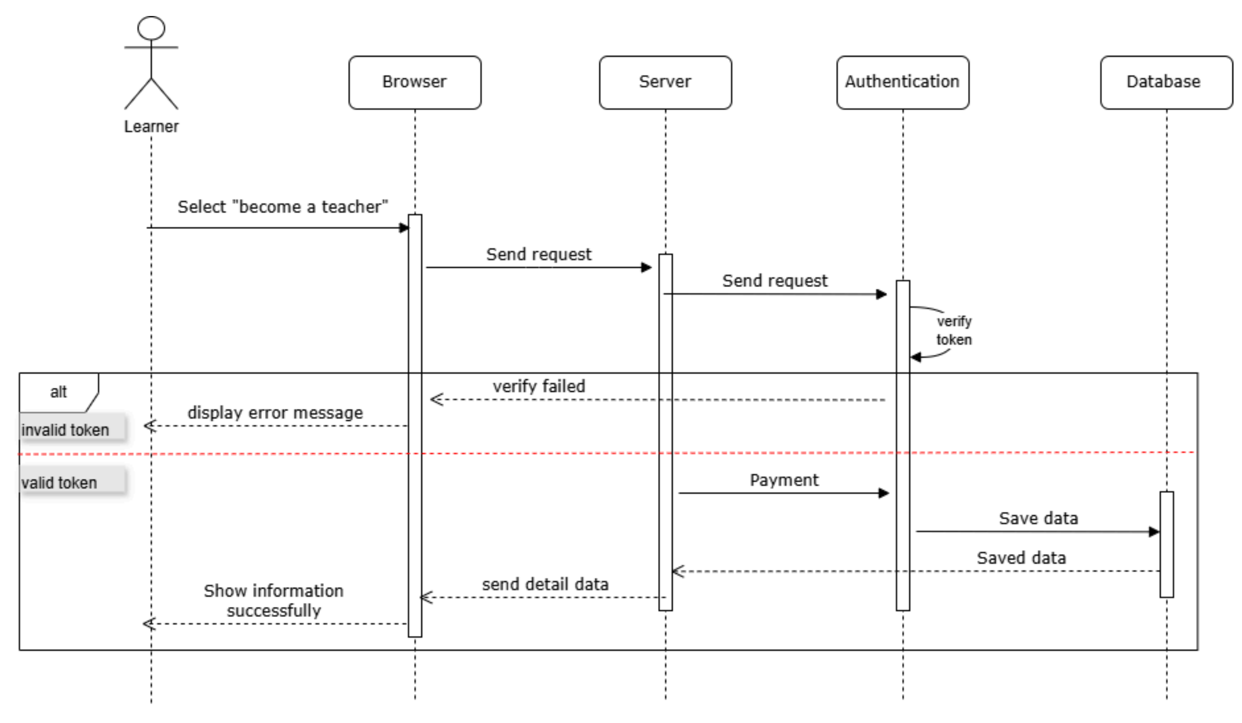
ID and Name	UC-L-5 : View and Edit personal account details
Date Created	16/01/2025

Primary Actor	Learner	N/A	N/A
Description	A function that allows learners to view and update their personal account details, including name, email and other profile-related information.		
Trigger	A learner accesses the "My Account" section from the BrainBox learner portal.		
Preconditions	PRE-1. The learner must be logged into the system. PRE-2. The learner's account data must exist in the database.		
Postconditions	Success: POST-1. The updated account details are saved in the system and reflected in the learner's profile. Failure: POST-2. The system does not save the changes, and an error message is displayed explaining the issue.		
Normal Flow	NF-1. Edit personal account details 1. The learner logs into the system. 2. The learner navigates to the "My Account" section. 3. The system displays the learner's current account details. 4. The learner selects the "Edit" option. 5. The learner updates their account details (e.g., name, email, phone). 6. The learner saves the changes by clicking the "Save" button. 7. The system validates the input: - If valid, the system saves the updates and displays a confirmation message. - If invalid, the system displays an error message and highlights the invalid fields.		
Alternative Flow	Edit Cancelled: 1. The learner selects the "Edit" option but decides not to save changes. 2. The learner clicks "Cancel," and the system discards any unsaved changes.		
Exceptions	E-1. Session Timeout: If the learner's session expires during the process, the system redirects them to the login page.		
Priority	Medium		
Frequency of Use	Occasionally, depending on how often learners update their personal		

	details.
Business Rules	BR-001, BR-003, BR-004, BR-005, BR008 from SRS document

6. Learners become teachers

a) Sequence Diagram



UC-L-6 : Learners become teachers

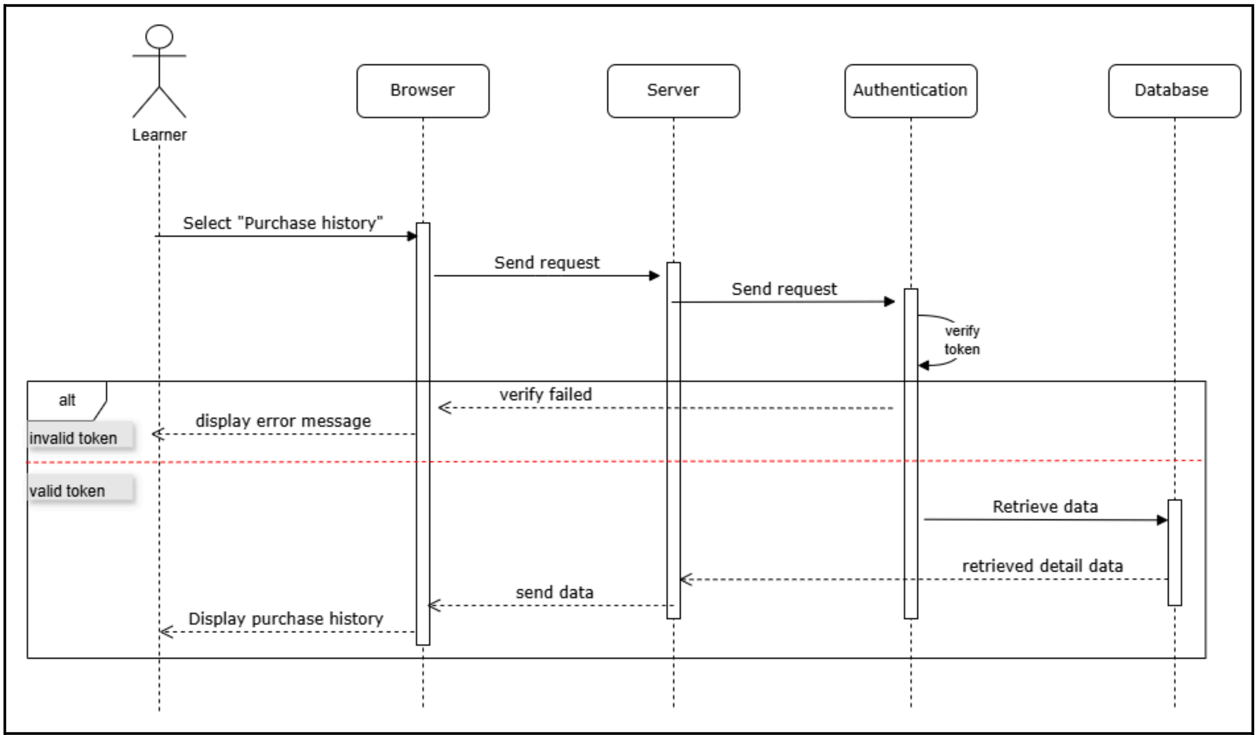
b) Flow Diagram

ID and Name	UC-L-6 : Learners become teachers		
Date Created	13/03/2025		
Primary Actor	Learner	N/A	N/A
Description	A function that allows learners to apply to become teachers on the BrainBox platform.		
Trigger	A learner accesses the "Become a Teacher" section from the BrainBox learner portal.		
Preconditions	PRE-1. The learner must be logged into the system. PRE-2. The learner must have an active account.		
Postconditions	Success:		

	POST-1. The learner's application is submitted successfully and is pending review. POST-2. If approved, the learner is granted teacher access and can create courses. Failure: POST-3. The system does not submit the application, and an error message is displayed explaining the issue.
Normal Flow	NF-1. Apply to become a teacher 1. The learner logs into the system. 1. The learner navigates to the "Become a Teacher" section. 2. The system displays the application form, payment. 3. The system validates the input: a. If valid, the system submits the application and notifies the admin for review. b. If invalid, the system displays an error message and highlights the invalid fields. 4. If approved, the learner receives confirmation and gains teacher access. 5. If rejected, the learner is notified with a reason for rejection.
Alternative Flow	Application Cancelled: 1. The learner starts the payment step. 2. The learner clicks "Cancel," and the system discards any unsaved changes.
Exceptions	E-1. Session Timeout: If the learner's session expires during the process, the system redirects them to the login page.
Priority	High
Frequency of Use	Occasionally, depending on how often learners apply to become teachers.
Business Rules	BR-001, BR-003, BR-008, BR-009 from SRS document

7. Learner purchase history

a) Sequence Diagram



UC-L-7 : Learner purchase history.

b) Flow Diagram

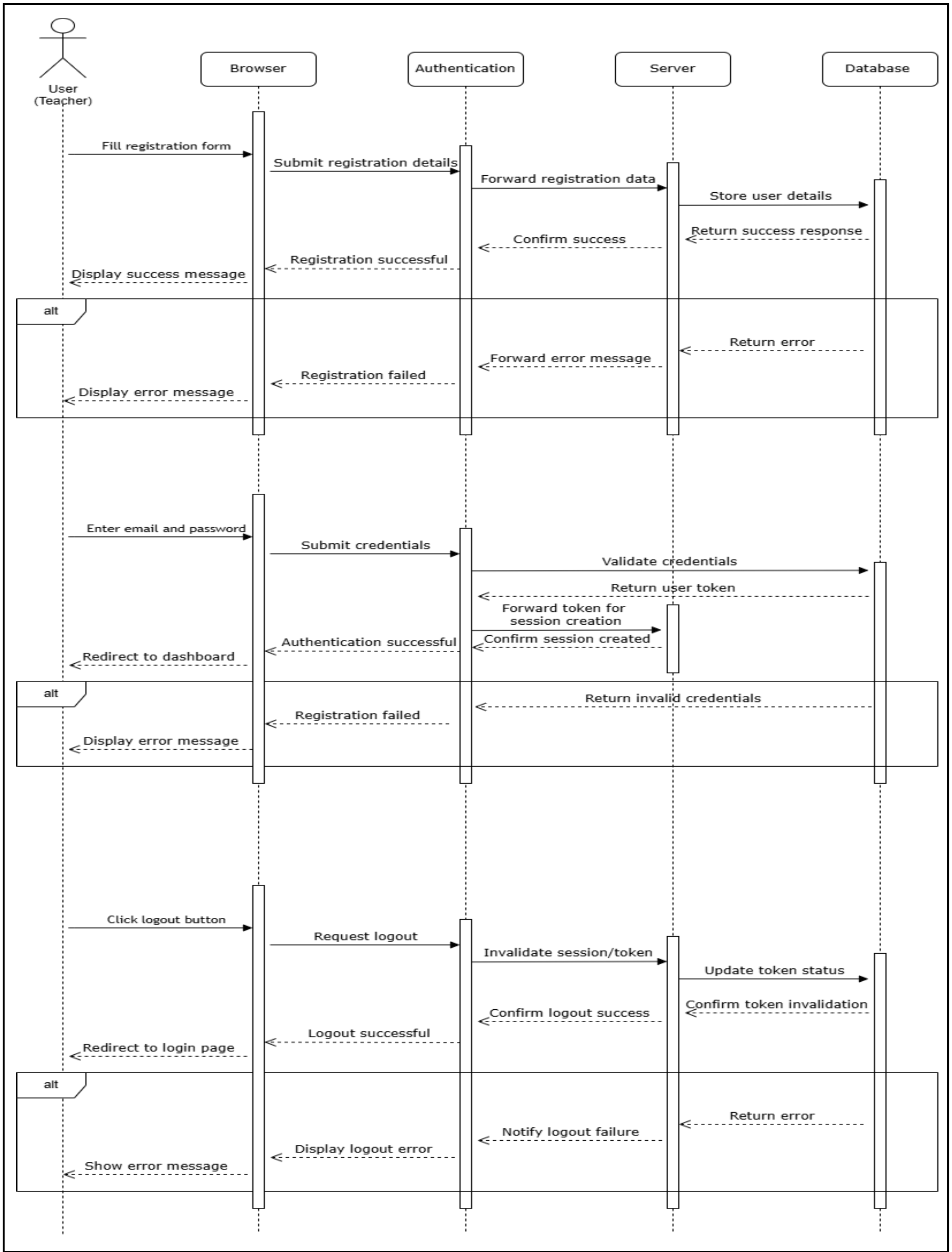
ID and Name	UC-L-7 : Learner purchase history.		
Date Created	13/03/2025		
Primary Actor	Learner	N/A	N/A
Description	A function that allows learners to view their past purchases on the BrainBox platform. The purchase history includes details such as course name, purchase date, amount paid, and payment method.		
Trigger	A learner accesses the "Purchase History" section from the BrainBox learner portal.		
Preconditions	PRE-1. The learner must be logged into the system. PRE-2. The learner must have completed at least one purchase.		
Postconditions	Success: POST-1. The system displays the learner's complete purchase history.		

	POST-2. The learner can view detailed information for each purchase. Failure: POST-3. If no purchases are found, the system displays a message indicating no purchase history is available.
Normal Flow	NF-1. View purchase history: 1. The learner logs into the system. 2. The learner navigates to the "Purchase History" section. 3. The system retrieves the learner’s past purchases from the database. 4. The system displays a list of purchased courses, including 5. The learner can select a specific purchase to view detailed information.
Alternative Flow	No Purchases Found: 1. The learner accesses the "Purchase History" section. 2. The system does not find any purchase records. 3. The system displays a message: "No purchase history available."
Exceptions	E-1. Session Timeout: If the learner's session expires during the process, the system redirects them to the login page.
Priority	Medium
Frequency of Use	Occasionally, depending on how often learners make purchases.
Business Rules	BR-001, BR-002, BR-003, BR-004, BR-005, BR-010 from SRS document

B. Teacher

1. Register, Login and Logout

a) Sequence Diagram


User
(Teacher)
Browser
Authentication
Server
Database
Fill registration form
Submit registration details
Forward registration data
Store user details
Return success response
Confirm success
Registration successful
Display success message
alt
Return error
Forward error message
Registration failed
Display error message
Enter email and password
Submit credentials
Validate credentials
Return user token
Forward token for
session creation
Confirm session created
Authentication successful
Redirect to dashboard
alt
Return invalid credentials
Registration failed
Display error message
Click logout button
Request logout
Invalidate session/token
Update token status
Confirm token invalidation
Confirm logout success
Logout successful
Redirect to login page
alt
Return error
Notify logout failure
Display logout error
Show error message

UC-T-1 : Login

b) Flow Diagram

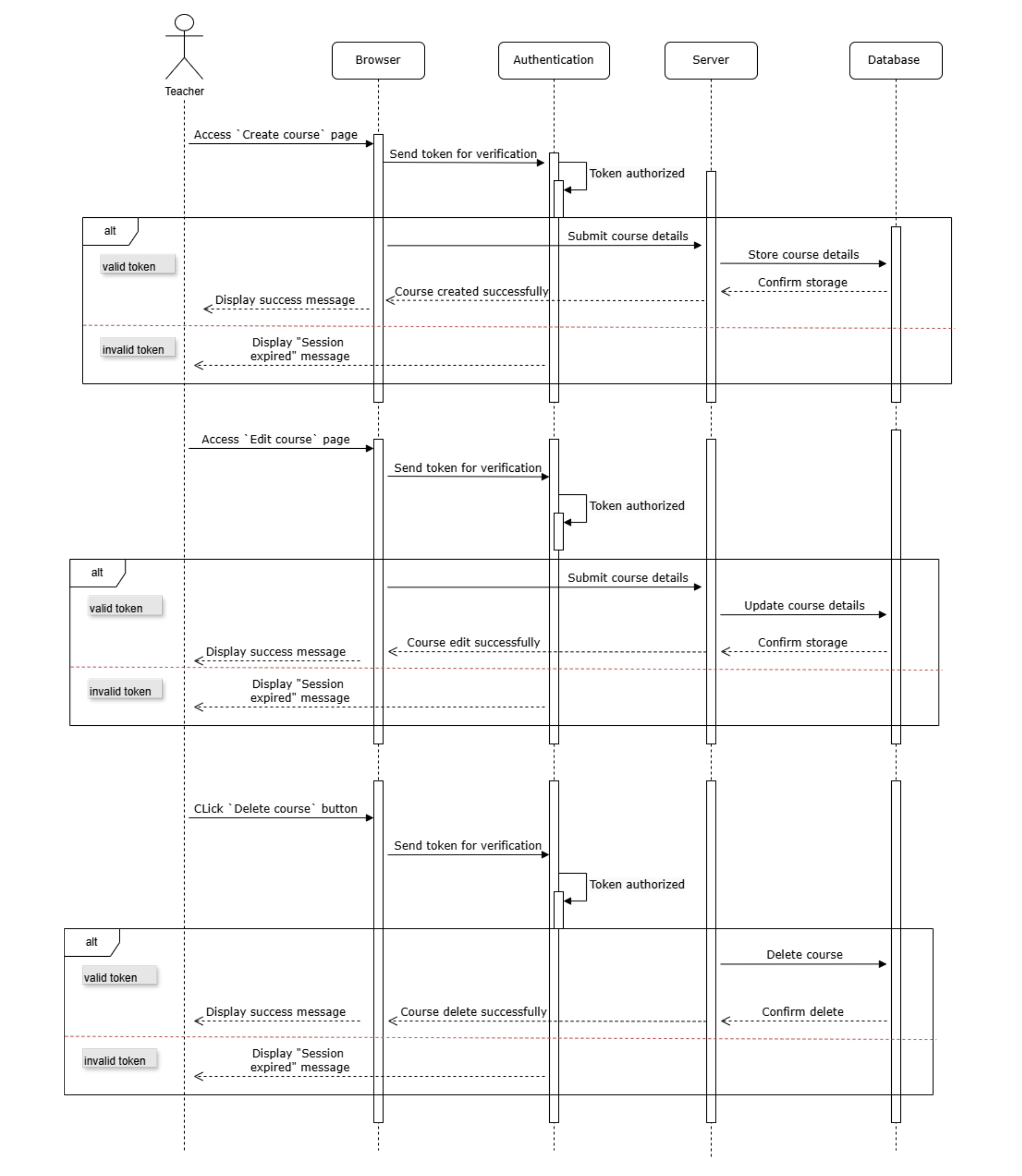
ID and Name	UC-T-1 : Register, Login and Logout (Clerk)		
Date Created	17/01/205		
Primary Actor	Teacher	N/A	N/A
Description	A function that allows teachers to register, log into the BrainBox system and log out of BrainBox.		
Trigger	A teacher accesses the BrainBox system through a web browser.		
Preconditions	1. The teacher has not logged in to the BrainBox. 2. The teacher must have valid credentials (username and password) registered in the system. 3. The system, including the authentication server and database, must be operational. 4. The teacher login or registration page must be accessible via the browser.		
Postconditions	Success: 1. The teacher successfully registers or logs into the system and is redirected to the teacher dashboard. 2. The teacher's name is displayed in the upper left corner of the dashboard. Failure: 1. The teacher remains on the login page, and an error message is displayed indicating the reason		
Normal Flow	**NF-1. Register** 1. The teacher opens the browser and accesses the registration page. 2. The browser displays the registration form. 3. The teacher fills in required details (e.g., username, password, email). 4. The browser sends the registration data to the Authentication Server. 5. The Authentication Server forwards the data to the Server. 6. The Server stores the details in the Database. 7. If the registration is successful, the Database confirms back to the Server.		

	8. The Server notifies the Authentication Server, which redirects the browser to a success page. **NF-2. Login** 1. The teacher opens the browser and accesses the login page. 2. The browser displays the login form. 3. The teacher enters their username and password. 4. The browser sends the credentials to the Authentication Server. 5. The Authentication Server validates the credentials with the Database. 6. If valid, the Authentication Server issues a session token and sends it to the Server. 7. The Server creates a session and confirms back to the Authentication Server. 8. The Authentication Server redirects the browser to the teacher dashboard. 9. The browser displays the teacher's name on the dashboard. **NF-3. Logout** 1. The teacher clicks the "Logout" button on the dashboard. 2. The browser sends a logout request to the Auth Server 3. The Authentication Server forwards the request to the Server. 4. The Server invalidates the session in the Database. 5. The Database confirms the session is invalidated. 6. The Authentication Server redirects the browser to the login page. 7. The browser displays a logout confirmation message.
Alternative Flow	**AF-1. Register** 1. The teacher enters invalid details (e.g., weak password, duplicate email). 2. The Authentication Server returns an error message to the browser. 3. The browser displays the error and prompts the teacher to correct the details. **AF-2. Login** 1. The teacher clicks "Forgot Password" on the login page. 2. The browser displays a password recovery form. 3. The teacher provides their registered email address. 4. The Authentication Server sends a password reset link to the email. 5. The teacher resets their password and logs in using the new

	credentials. **AF-3. Logout** 1. If the logout request fails (e.g., due to system errors), the teacher remains logged in and sees an error message.
Exceptions	**E-1. Register** 1. System Unavailable: If the Database or Authentication Server is down, the registration cannot be completed. 2. Duplicate Account: The system prevents duplicate registration for the same email/username. **E-2. Login** 1. Invalid Credentials: The Authentication Server rejects the login attempt and returns an error message to the browser. 2. Account Locked: If the teacher enters incorrect credentials multiple times, the account is locked, requiring manual intervention. **E-2. Logout** 1. Token Expired: If the session token is no longer valid, the system logs the teacher out automatically.
Priority	High
Frequency of Use	Daily, depending on the number of teachers accessing the system.
Business Rules	BR001, BR003, BR004, BR005, BR008 from SRS document

2. Create, Edit and Delete new courses with multimedia content.

a) Sequence Diagram



UC-T-2 : Create, Edit and Delete new courses with multimedia content.

b) Flow Diagram

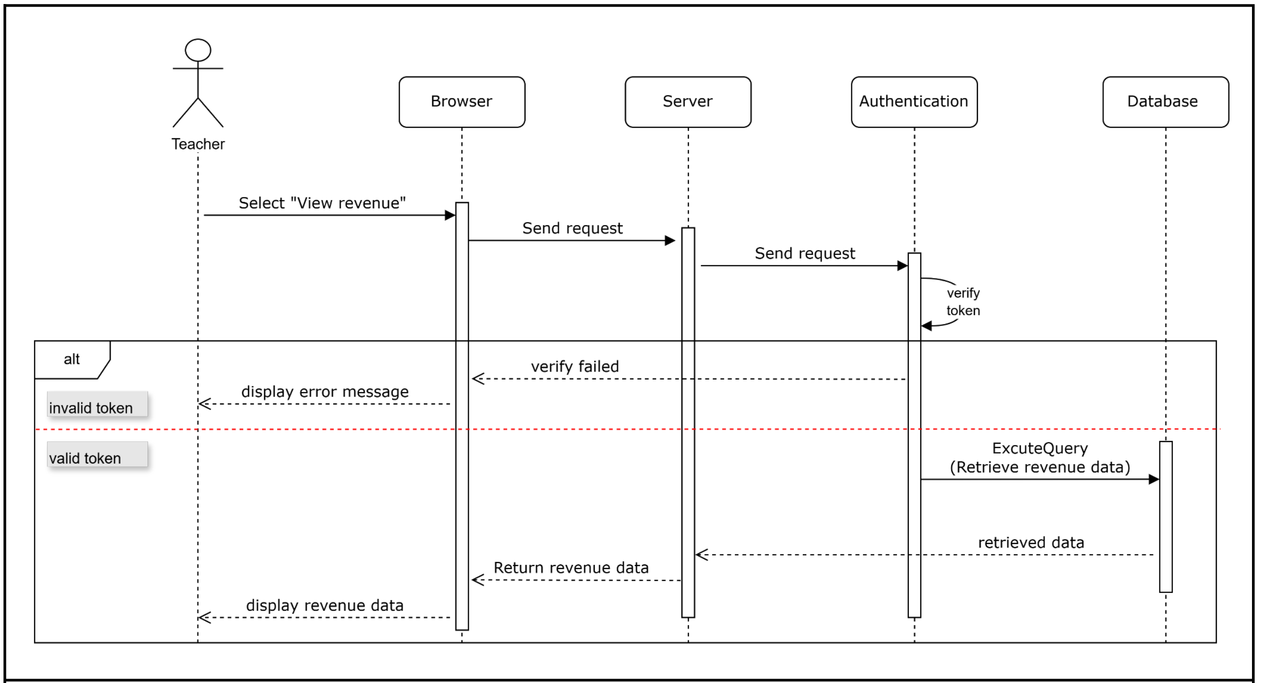
ID and Name	UC-T-2 : Create, Edit and Delete new courses with multimedia content.		
Date Created	20/01/2025		
Primary Actor	Teacher	Secondary Actor	N/A
Description	A function that allows teachers to Create, Edit and Delete new courses with multimedia content.		
Trigger	A teacher clicks the button Create, Edit or Delete course through a web browser.		
Preconditions	**PRE-1**. The user must be logged in with a valid teacher account. **PRE-2**. The teacher's account must be verified and active. **PRE-3**. The platform must support multimedia upload (video, audio, PDF, images). **PRE-4**. The course category and pricing information must be predefined in the system.		
Postconditions	**POST-1**. A new course is created, edited, or deleted, and changes are reflected in the system database. **POST-2**. The system sends a confirmation notification to the teacher via email or messaging app (e.g., Zalo). **POST-3**. The multimedia content associated with deleted courses is also removed from storage.		
Normal Flow	**NF-1. The teacher navigates to the "Manage Courses" section in their dashboard.** **NF-2. The teacher selects the desired action: Create, Edit, or Delete.** **NF-3. For creating a course:** 1. The teacher provides course details (title, description, price, category, and duration). 2. The teacher uploads multimedia content (e.g., videos, PDFs, images). 3. The teacher clicks "Publish" to submit the course for admin review. **NF-4. For editing a course:** 1. The teacher selects the course they want to edit. 2. The teacher modifies the course details or replaces multimedia		

	content. 3. The teacher clicks "Update" to save changes. **NF-5. For deleting a course:** 1. The teacher selects the course to delete. 2. The system displays a confirmation dialog to avoid accidental deletion. 3. The teacher confirms the deletion, and the system removes the course.
Alternative Flow	**AF-1. If the teacher attempts to create or edit a course without uploading multimedia content:** 1. The system displays an error message: "At least one multimedia file is required." **AF-2. If the teacher attempts to delete a course currently enrolled by learners:** 1. The system restricts deletion and displays a message: "This course cannot be deleted as it has active enrollments." 2. The teacher can choose to unpublish the course instead.
Exceptions	**E-1. The system encounters a storage error during multimedia file upload:** 1. The system displays an error message: "Failed to upload multimedia content. Please try again." **E-2. Network connectivity is lost during any action:** 1. The system displays a message: "Action failed due to network issues. Please reconnect and try again." **E-3. Unauthorized access attempt by a non-teacher user:** 1. The system logs the incident and redirects the user to an error page with the message: "You do not have permission to manage courses." **E-4. The course title or description exceeds character limits:**

	1. The system displays a message: "Course title/description exceeds the maximum allowed length. Please shorten the content."
Priority	High
Frequency of Use	Daily, depending on the number of teachers accessing the system.
Business Rules	BR001, BR003, BR004, BR005, BR008 from SRS document

3. View revenue reports for their courses.

a) Sequence Diagram



UC-T-3: View revenue reports for their courses

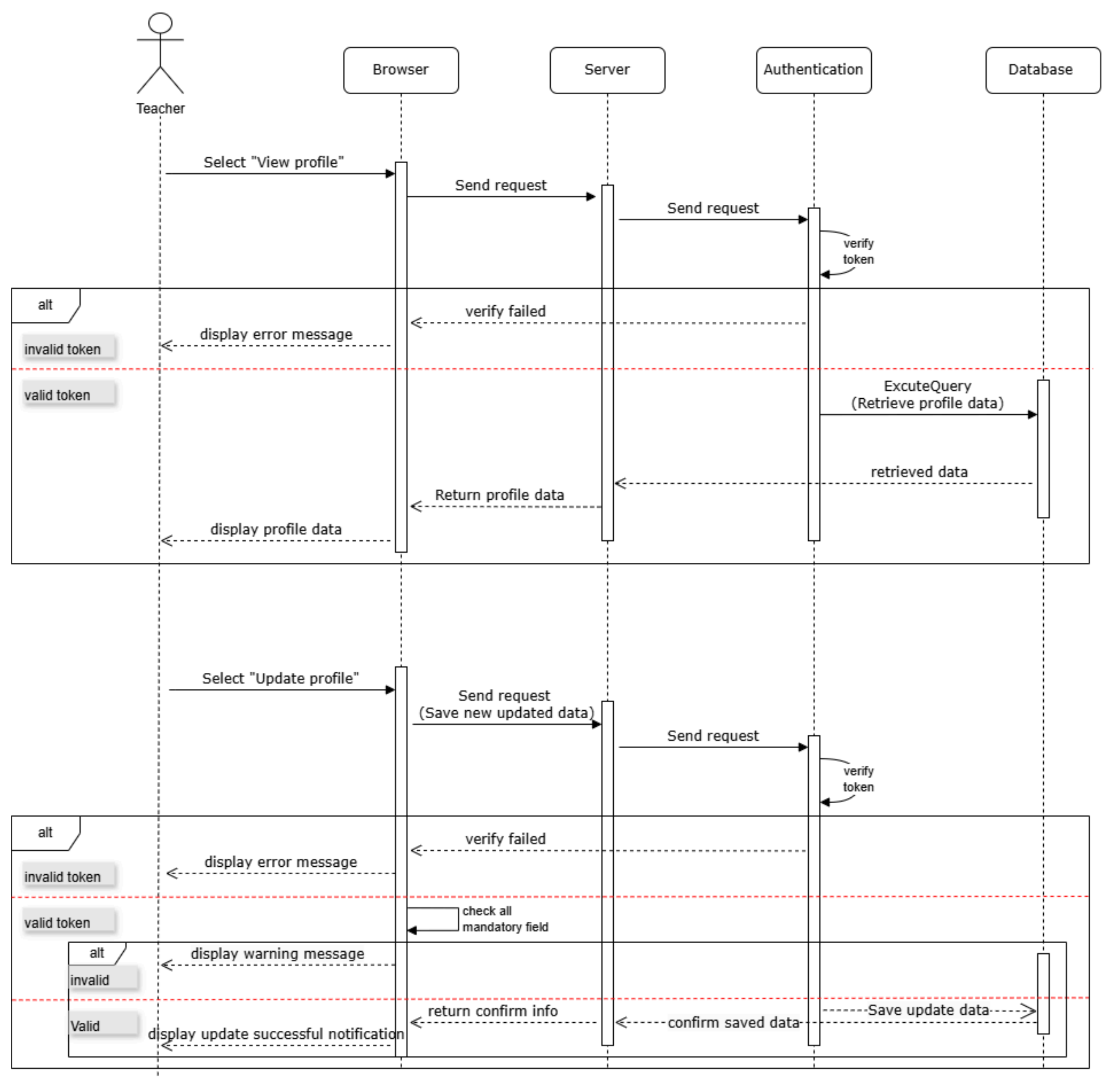
b) Flow Diagram

ID and Name	UC-T-3: View revenue reports for their courses		
Date Created	26/01/2015		
Primary Actor	Teacher	Secondary Actors	BrainBox System
Description	A function that allows teachers to view revenue reports for their courses.		
Trigger	The Teacher clicks the `View revenue` button to view revenue reports for their courses.		

Preconditions	1. The teacher must be logged into the BrainBox System.
Postconditions	1. Data Displayed: The revenue reports are correctly displayed.
Normal Flow	**NF-1. View the revenue reports** 1. Click the \`View revenue\` button.
Alternative Flow	N/A
Exceptions	**E-1. Database connection failure** 1. The BrainBox System generates an exception indicating the database connection failure. 2. The Teacher is notified of the issue and prompted to check network connectivity or contact technical support.
Priority	High
Frequency of Use	Approximately 10 teachers, each one average of 20 usage per day.
Business Rules	BR-010 from SRS document

4. Manage personal account information (profile, contact details)

a) Sequence Diagram



UC-T-4: Manage personal account information

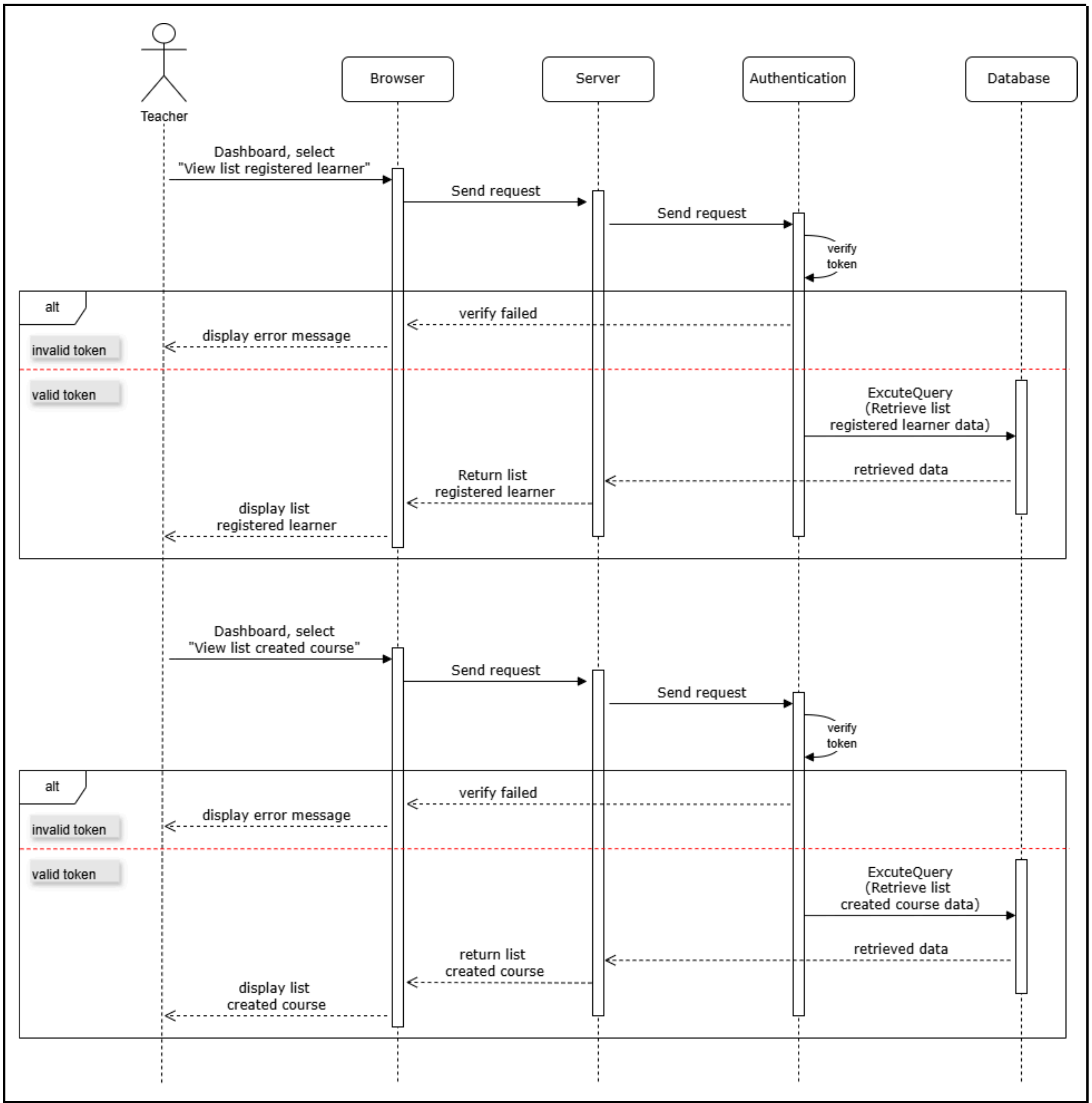
b) Flow Diagram

ID and Name	UC-T-4: Manage personal account information		
Date Created	17/01/2015		
Primary Actor	Teacher	Secondary Actors	BrainBox System
Description	A function that allows teachers to manage their accounts by viewing and updating their profile, and contact details.		
Trigger	The Teacher clicks the `View profile` button to view and then clicks the		

	\`Save\` button to update their profile.
Preconditions	2. The teacher must be logged into the BrainBox System.
Postconditions	2. Data Displayed: The teacher’s profile is correctly displayed. 3. The BrainBox system updates the teacher’s profile in the database. 4. The website displays a notification to inform the teacher that the profile has been successfully updated.
Normal Flow	**NF-1. View the teacher’s profile** 2. Click the \`View\` profile button. **NF-2. Update the teacher’s profile** 1. Click the \`Update\` button. 2. Change profile information 3. Click the \`Save\` button. 4. The system checks if all mandatory fields have been filled: a. If Yes, go to the next step. b. If No, go to E-2. 5. The system saves new data into the database and displays a successful pop-up message “Update profile successfully.”
Alternative Flow	N/A
Exceptions	**E-1. Database connection failure** 3. The BrainBox System generates an exception indicating the database connection failure. 4. The Teacher is notified of the issue and prompted to check network connectivity or contact technical support. **E-2. Update the teacher’s profile** 1. Display warning pop-up message “Please fill all mandatory fields.”.
Priority	High
Frequency of Use	Approximately 10 teachers, each one average of 20 usage per day.
Business Rules	BR-001, BR-005, BR-008, BR-009.

5. Dashboard

a) Sequence Diagram


Teacher
Browser
Server
Authentication
Database
Dashboard, select "View list registered learner"
Send request
Send request
verify token
alt
invalid token
verify failed
display error message
valid token
ExcuteQuery (Retrieve list registered learner data)
retrieved data
Return list registered learner
display list registered learner
Dashboard, select "View list created course"
Send request
Send request
verify token
alt
invalid token
verify failed
display error message
valid token
ExcuteQuery (Retrieve list created course data)
retrieved data
return list created course
display list created course

UC-T-5 : Dashboard

b) Flow Diagram

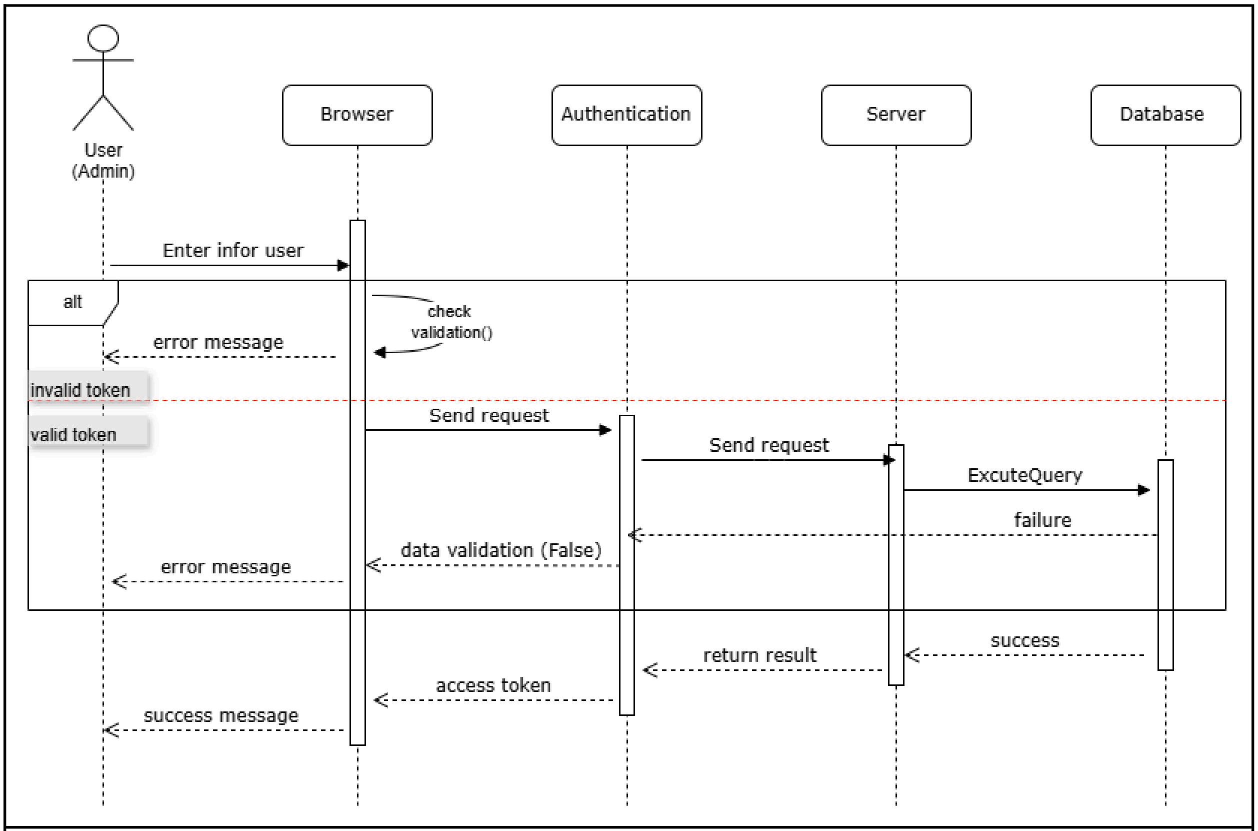
ID and Name	UC-T-5 : Dashboard
Date Created	21/01/2025

Primary Actor	Teacher	N/A	N/A
Description	A dashboard feature that allows teachers to view a list of learners registered for their courses and a list of courses they have created. The dashboard provides an overview of learner details and course statistics.		
Trigger	A teacher logs into the system and accesses the dashboard page.		
Preconditions	PRE-1. The teacher must have an active account and be logged in.		
Postconditions	Success: POST-1. The teacher can view a list of learners registered for their courses and a list of their created courses. Failure: POST-2. If no data is available, the dashboard displays appropriate messages (e.g., "No registered learners" or "No courses created yet").		
Normal Flow	NF-1. The teacher logs into the system. NF-2. The system displays the dashboard with the following sections: - List of Registered Learners: Displays learner details such as name, email, registered course, and registration date. - List of Created Courses: Displays course details such as course title, creation date, and the number of registered learners. NF-3. The teacher can: - Filter the list of learners by course. - Sort the list of learners or courses by various fields (e.g., name, date).		
Alternative Flow	Export Data (Optional): - The teacher selects an option to export the learner or course data (e.g., to a CSV file). - The system generates the file and allows the teacher to download it.		
Exceptions	E-1. Session Timeout: If the teacher's session expires while on the dashboard, the system redirects them to the login page.		
Priority	High		
Frequency of Use	Daily or as needed, depending on the teacher's activities.		
Business Rules	BR-001, BR-002, BR-003, BR-004, BR-005, BR-006, BR-007 from SRS document		

C. Admin

1. Log in

a) Sequence Diagram



UC-A-1 : Login

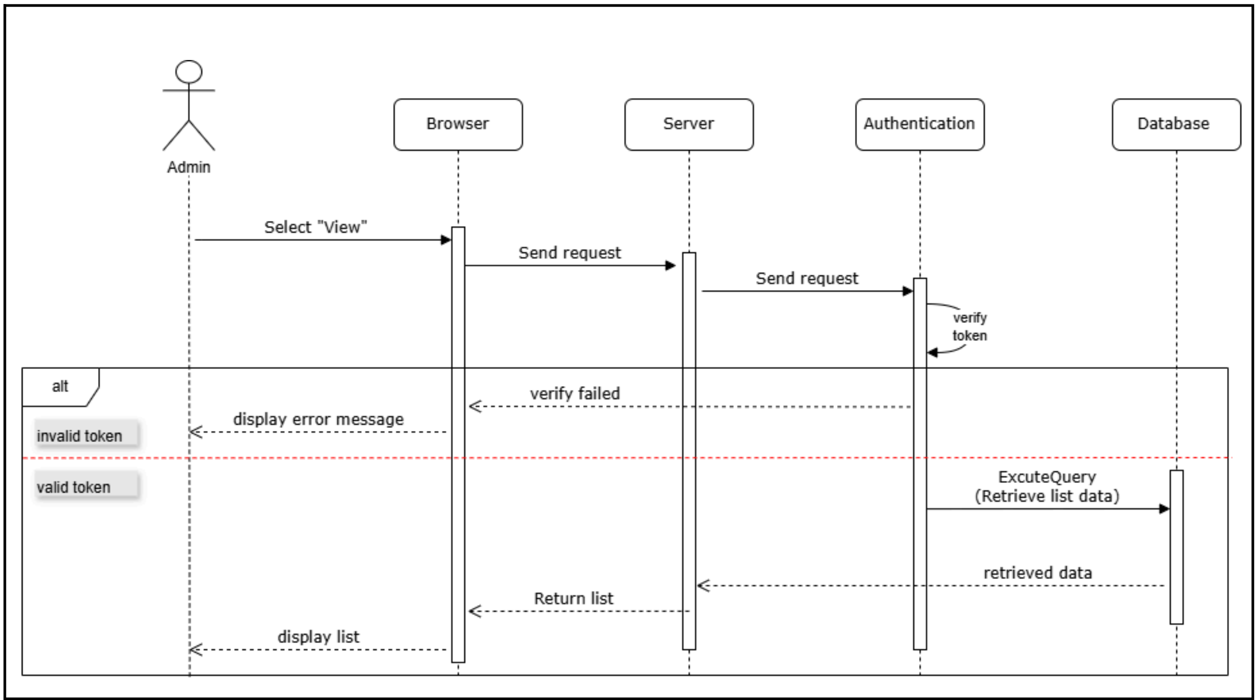
b) Flow Diagram

ID and Name	UC-A-1 : Login		
Date Created	16/01/205		
Primary Actor	Administrator	N/A	N/A
Description	A function that allows administrators to log into the BrainBox system.		
Trigger	An administrator accesses the BrainBox system.		
Preconditions	PRE-1. Administrator has not logged in to the BrainBox. PRE-2. The administrator must have valid credentials (username and password) registered in the system. PRE-3. The system must be operational and connected to the database. PRE-4. Admin login page must be accessible.		

Postconditions	Success: POST-1. The administrator is logged in and redirected to the admin dashboard. POST-2. The Employee's name will be displayed in the upper left corner of the Dashboard according to the account they login to. POST-3. Admin name will be displayed on Dashboard according to the account they are logged in with. Failure: POST4. The administrator remains on the login page, and an error message is displayed indicating the reason
Normal Flow	NF-1. Log in to the website 1. The administrator accesses the login page. 2. The system displays the login form. 3. The administrator enters their username and password. 4. The administrator submits the login form. 5. The system validates the entered credentials: - If valid, the system logs the administrator in and redirects them to the admin dashboard. - If invalid, the system displays an error message and prompts the administrator to try again.
Alternative Flow	1. The administrator clicks on the "Forgot Password" link. 2. The system displays a password recovery form. 3. The administrator provides their registered email address. 4. The system sends a password reset link to the email. 5. The administrator resets their password and logs in using the new credentials.
Exceptions	E-1. Invalid Credentials: The system rejects the login attempt and displays an error message. E-2. Account Locked: If the administrator enters incorrect credentials multiple times, their account may be locked, requiring manual intervention to unlock.
Priority	High
Frequency of Use	Daily, depending on the number of administrators accessing the system.
Business Rules	BR-001, BR-003, BR-004, BR-005, BR-008 from SRS document

2. Manage user accounts

a) Sequence Diagram



UC-A-2: Manage user accounts

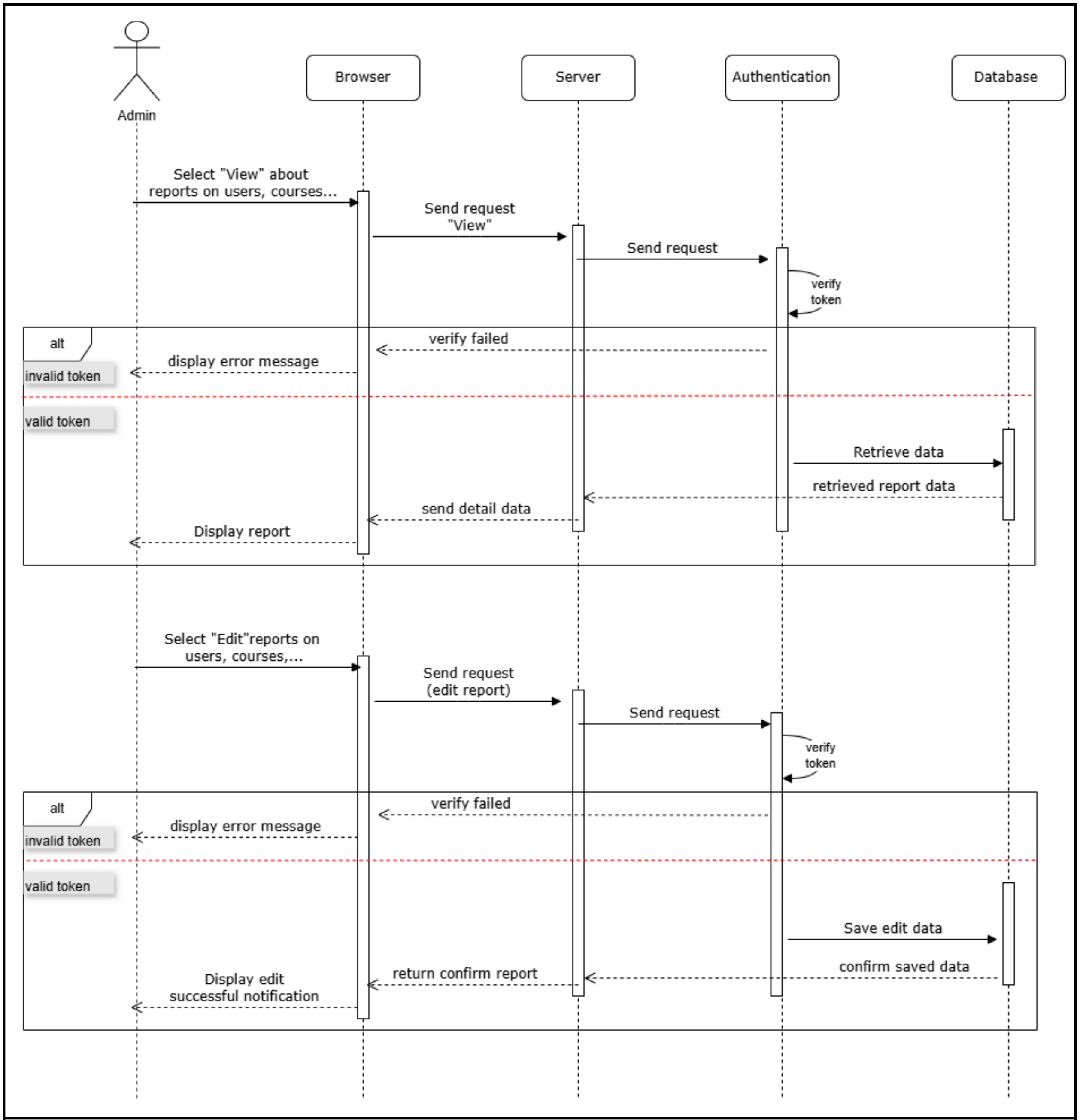
b) Flow Diagram

ID and Name	UC-A-2: Manage user accounts		
Date Created	17/01/2015		
Primary Actor	Administrator	Secondary Actors	BrainBox System
Description	A function that allows administrators to manage user accounts by viewing, searching, and sorting in the BrainBox system.		
Trigger	An administrator clicks the `Teachers` button or `Learners` button.		
Preconditions	An administrator must be logged into the BrainBox System. There are already customers in the database.		
Postconditions	1. Data Displayed: All teachers of learners's information is correctly displayed.		
Normal Flow	**NF-1. View all teachers or learners** 1. Click the `Teachers` button or the `Learners` button.		

	NF-2. Search teachers or learners 1. Search teachers or learners in real time by input search. **NF-3. Sort teachers or learners** 1. Sort teachers or learners by categories.
Alternative Flow	N/A
Exceptions	**E-1. Database connection failure** 1. The BrainBox System generates an exception indicating the database connection failure. 2. An administrator is notified of the issue and prompted to check network connectivity or contact technical support.
Priority	High
Frequency of Use	Only 1 administrator, an average of 20 usages per day.
Business Rules	BR-010

3. View, edit and analyze reports on users, courses, and payments.

a) Sequence Diagram



UC-A-3: View, edit and analyze reports on users, courses, and payments

b) Flow Diagram

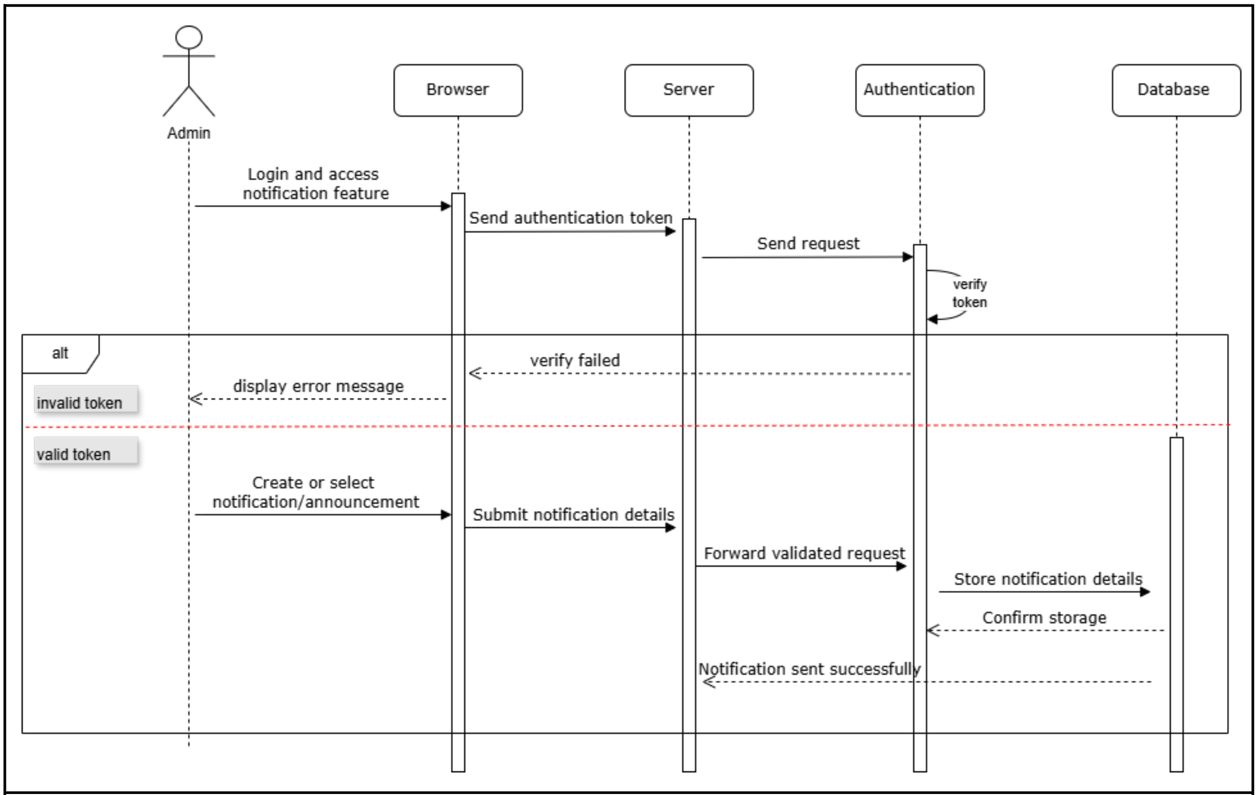
ID and Name	UC-A-3: View, edit and analyze reports on users, courses, and payments		
Date Created	23/01/2025		
Primary Actor	Administrator	Secondary Actors	BrainBox System
Description	A feature that allows administrators to view, edit, and analyze detailed		

	reports on system users, courses, and payment activities. This functionality helps admins monitor system performance and take necessary actions based on insights from the reports.
Trigger	An admin accesses the "Reports" section of the BrainBox system.
Preconditions	PRE-1. The admin must be logged into the system with the appropriate access rights. PRE-2. The system must have sufficient data on users, courses, and payments.
Postconditions	POST-1. The admin views, edits, or generates analytical insights from the reports on users, courses, and payments.
Normal Flow	1. The admin logs into the system and navigates to the "Reports" section. 2. The system displays a dashboard with three main reporting options: ● Users Report: List of all users (learners and teachers) with details such as name, email, registration date, activity status, etc. ● Courses Report: List of all courses with details such as course title, creation date, number of learners enrolled, and course status. ● Payments Report: List of payment transactions with details such as payment method, amount, discount applied, and transaction date. 3. The admin can: ● Apply filters to reports (e.g., by date, status, or category). ● Sort data by fields (e.g., most recent payments or top-enrolled courses). ● Edit specific entries (e.g., update user activity status or correct course details). ● Generate charts and graphs for data analysis (e.g., revenue trends or user growth). ● Export reports to a file (e.g., CSV or PDF).
Alternative Flow	No Data Available: 1. If there is no data available for a report, the system displays a message such as "No data available for the selected filters."

Exceptions	E-1. Data Loading Error: If the system fails to load the report data, an error message is displayed, and the admin can retry. E-2. Unauthorized Access: If the admin does not have the required permissions to access or edit certain data, the system displays an "Access Denied" message.
Priority	High
Frequency of Use	Regularly, depending on the admin's need to monitor and manage system performance.
Business Rules	BR-001, BR-002, BR-003, BR-004, BR-005, BR-006, BR-008, BR-009 from SRS document

4. Send notifications or announcements to users.

a) Sequence Diagram


Admin
Browser
Server
Authentication
Database
Login and access notification feature
Send authentication token
Send request
verify token
alt
verify failed
display error message
invalid token
valid token
Create or select notification/announcement
Submit notification details
Forward validated request
Store notification details
Confirm storage
Notification sent successfully

UC-A-4: Send notifications or announcements to users

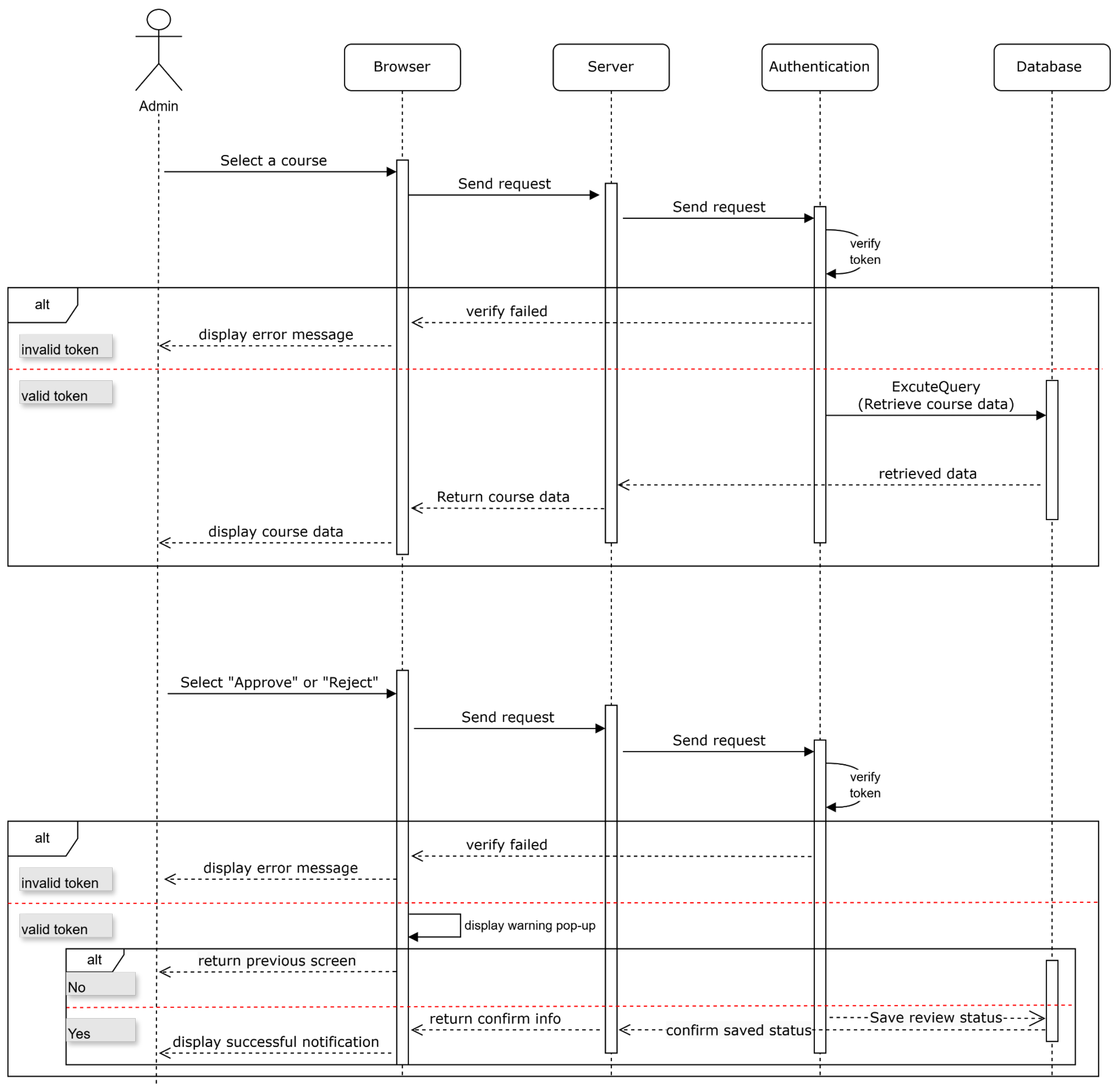
b) Flow Diagram

ID and Name	UC-A-4: Send notifications or announcements to users		
Date Created	19/01/2015		
Primary Actor	Administrator	Secondary Actors	BrainBox System
Description	This functionality allows authorized users (e.g., administrators or teachers) to send notifications or announcements to a group of users (e.g., students, parents, or other staff). Notifications can be about upcoming events, updates, or reminders and can be delivered via in-app notifications or emails.		
Trigger	The user logs into the system and initiates the "Send Notification" action from the notification management page.		
Preconditions	PRE-1. The user must be authenticated and authorized to send notifications. PRE-2. The system must have a list of target users (e.g., students or parents) stored in the database. PRE-3. The system must have a working notification service (e.g., email or push notification service).		
Postconditions	POST-1. Notifications or announcements are created and successfully sent to the selected users. POST-2. Notification delivery status is logged in the database for tracking purposes.		
Normal Flow	1. The user logs into the system and accesses the notification feature. 2. The system verifies the user's authentication and permissions. 3. The user enters the notification details (e.g., message content, target audience, and delivery method). 4. The system validates the input and processes the request. 5. The server stores the notification in the database. 6. Notifications are sent to the target users via the selected channels (e.g., push notifications or email). 7. The user is notified of the successful delivery.		
Alternative Flow	**Invalid Token**: 1. If the user’s session has expired, the system displays an error message. 2. The user is redirected to the login page.		

	Incomplete Notification Details: 1. If mandatory fields (e.g., message content or target audience) are missing, the system prompts the user to complete the details.
Exceptions	1. **Database Error**: If the server fails to store the notification, an error message is displayed, and the user is asked to try again later. 2. **Notification Service Failure**: If the delivery system (e.g., email server) is down, the system logs the error and notifies the user of the issue. 3. **Invalid Target Audience**: If the target user list is empty or invalid, the system displays an error.
Priority	High
Frequency of Use	Daily or as needed, depending on the volume of announcements or notifications.
Business Rules	BR-001, BR-002, BR-003, BR-004, BR-005, BR-006 from SRS document

5. Approve or reject actions of the teacher of course.

a) Sequence Diagram



UC-A-5: Approve or reject actions of the teacher of course

b) Flow Diagram

ID and Name	UC-A-5: Approve or reject actions of the teacher of course		
Date Created	21/01/2015		
Primary Actor	Administrator	Secondary Actors	BrainBox System
Description	A function that allows administrators to manage courses by approving, rejecting after teachers modify courses in the BrainBox system.		
Trigger	An administrator clicks the `Review courses` button.		

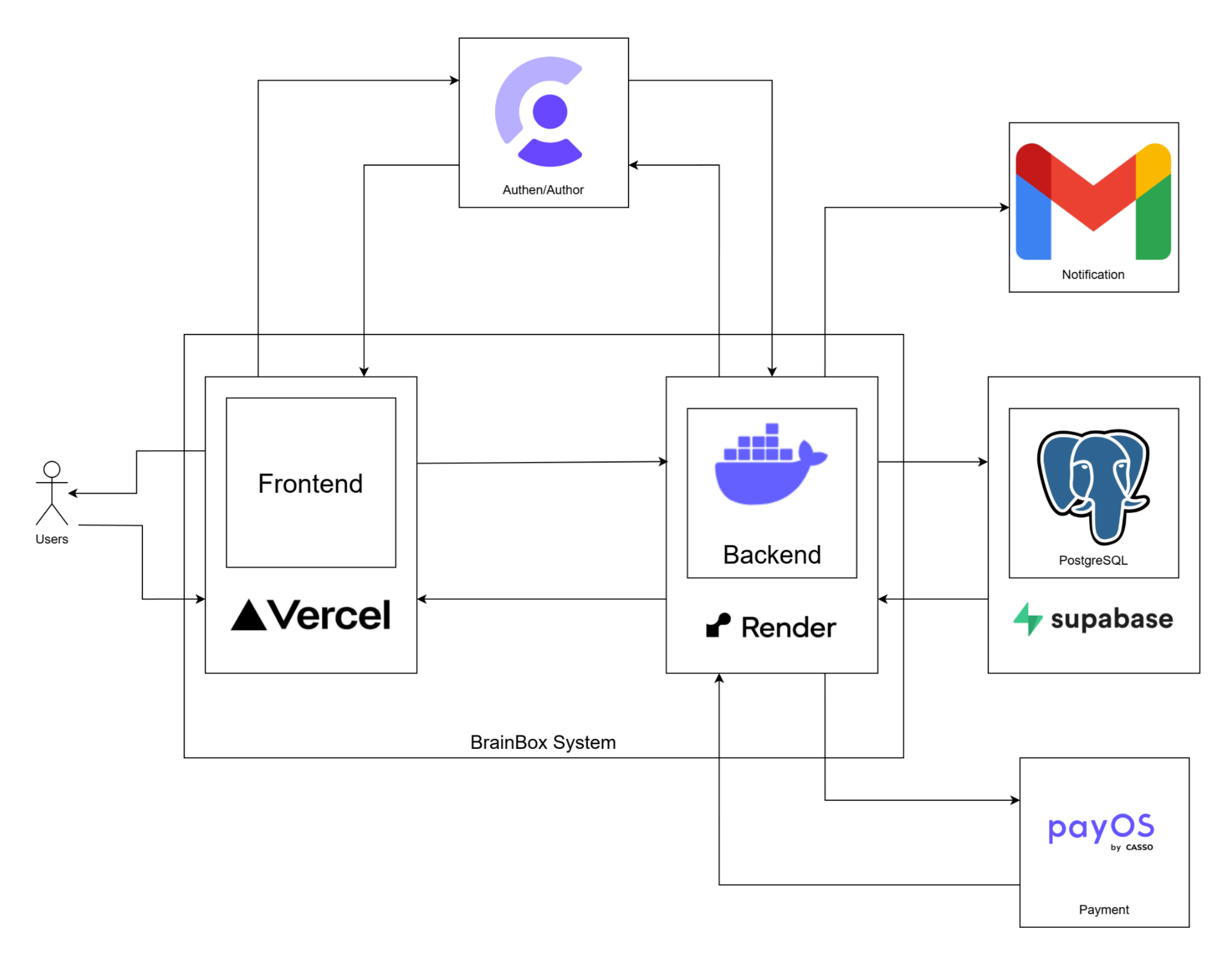
Preconditions	1. An administrator must be logged into the BrainBox System. 2. Click the `Review courses` button. 3. Select a course to review.
Postconditions	1. Courses were approved or rejected by the administrator.
Normal Flow	**NF-1. Approve or reject courses** 1. Click the `Approve` button or the `Reject` button. 2. Display pop-up to confirm again. a. If No, return list course to review. b. If Yes, go to the next step. 3. The system saves new status into the database.
Alternative Flow	N/A
Exceptions	**E-1. Database connection failure** 1. The BrainBox System generates an exception indicating the database connection failure. 2. An administrator is notified of the issue and prompted to check network connectivity or contact technical support.
Priority	High
Frequency of Use	Only 1 administrator, an average of 30 usages per day.
Business Rules	BR-010

III. UI/UX Design

Figma:

Link Figma: BrainBox Figma Design

IV. Deployment Diagram



IV. Deployment diagram