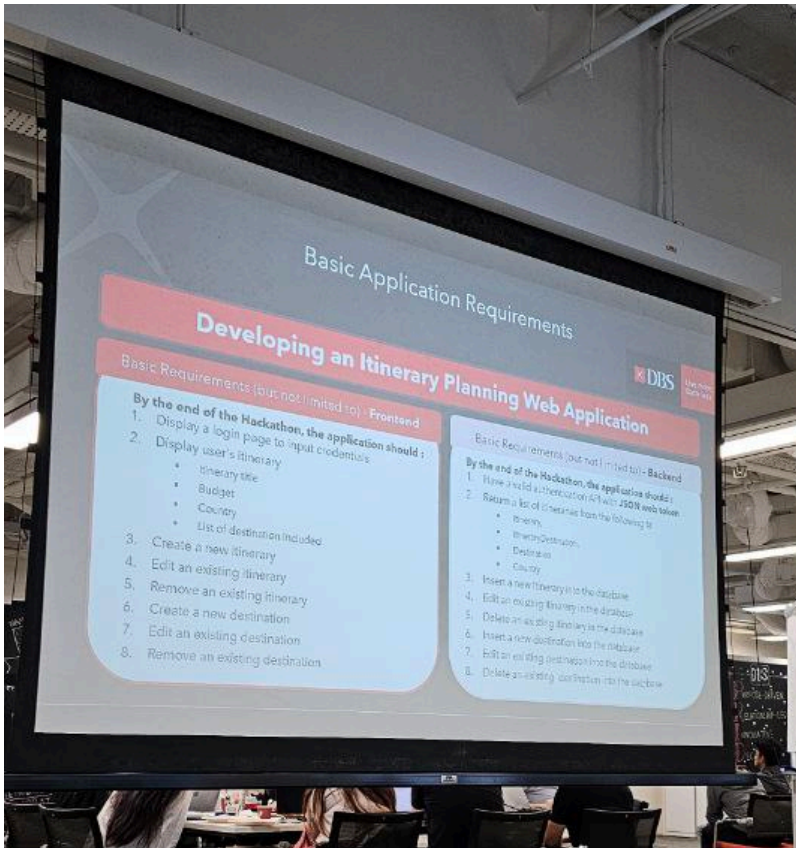

Itinerary Planning Web Application

Checklist (highlight when done?):

1) Login Page
2) Display User itinerary
    a) Itinerary title
    b) Budget
    c) Country
    d) List of destination included
3) Create a new itinerary
4) Edit an existing itinerary
5) Remove an existing itinerary
6) Create a new destination
7) Edit an existing destination
8) Remove an existing destination

Module		Basic Requirements (Frontend)	Basic Requirements (Backend)
Login	[1]	User must be able to login	Server must be able to authenticate a user's identity, using JSON Web Tokens for authentication
Dashboard	[2]	Display the itineraries created by the user with the following details: • Itinerary Title • Budget • Country • List of Destinations included	Returns a list of itineraries from the ***Itinerary, ItineraryDestination, Destination & Country*** table
Itinerary	[3]	Create new itinerary	Insert itinerary chosen from frontend into ***Itinerary*** table
	[4]	Edit existing itinerary	Ensure the ***Itinerary*** table is updated for each change
	[5]	Remove existing itinerary	Delete itinerary from the ***Itinerary*** table
Destination	[6]	Create new destination	Insert destination chosen from frontend into ***Destination*** table
	[7]	Edit existing destination	Ensure the ***Destination*** table is updated for each change
	[8]	Remove existing destination	Delete destination from the ***Destination*** table

Data Contract

Serial No	Endpoint	Description
1.	POST /api/auth/register	Register a user: First_name, last_name, username,password
2	POST /api/auth/login	Login a user

3	GET /api/:id/details	Dashboard Details
4	POST /api/itinerary	Create Itinerary
5	GET /api/itinerary	Get all Itineraries
6	GET /api/itinerary/:id	Get 1 Itinerary based on ID
7	PUT /api/itinerary/:id	Update 1 Itinerary based on ID
8	DELETE /api/itinerary/:id	Delete 1 Itinerary based on ID
8a	GET /api/destinations	Get all destinations
9	POST /api/destination	Create destination
10	UPDATE /api/destination/:id	Update destination
11	DELETE /api/destination/:id	Delete destination
12	GET /api/countries	Get all countries

Green means done

Interface:

API	frontend	backend
POST /api/auth/register	{first_name, last_name, username, password}	Return 201 Validation: ● First, last name is not empty ○ If not 401 ● First, last name is shorter than 50 ○ If not 401
POST /api/auth/login	{username, password}	Return 201 Validation: ● Username exists ○ If not 401 ● Password longer than 8 characters, shorter than 20 characters, contains at least one upper case and lower case letter and one digit ○ If not 401
POST api/auth/logout	{}	Return {“is_success” : True}
GET /api/:id/details	user_id from the url	Return { {itinerary 1, budget, country, [destinations]}, {itinerary 2, budget, country,

		[destinations]}, …, {itinerary n, budget, country, [destinations]} } }
POST /api/itinerary	{country_id, name, cost, notes}	Return 201 Validation: Return 500 if an error occurs while creating the itinerary
PUT /api/itinerary/:id	user_id from the url{country_id, name, cost, notes}	Return 200 Validation: ● 404 if itinerary not found ● 500 if an error occurs while creating the itinerary
DELETE /api/itinerary/:id	user_id from the url	Return 200 Validation: ● 404 if itinerary not found ● 500 if an error occurs while creating the itinerary
POST /api/destination	{country_id, user_id, budget, title}	Return 201 Validation: ● 500 if an error occurs while creating the destination
PUT /api/destination/:id	user_id from the url{country_id, user_id, budget, title}	● 404 if destination not found
DELETE /api/destination/:id	user_id from the url	Return 200 Validation: ● 404 if destination not found