Institute of Technology of Cambodia

Department of Information and Communication Engineering

Assignment TP10

TP10: VueJS, NodeJS Authentication (Continue)

Group: I4 GIC(C)

Lecturers	Subjects
HOK TIN	TP Internet-Programming II(IP)
Student's Name:	ID
ROTHA DAPRAVITH	e20190915

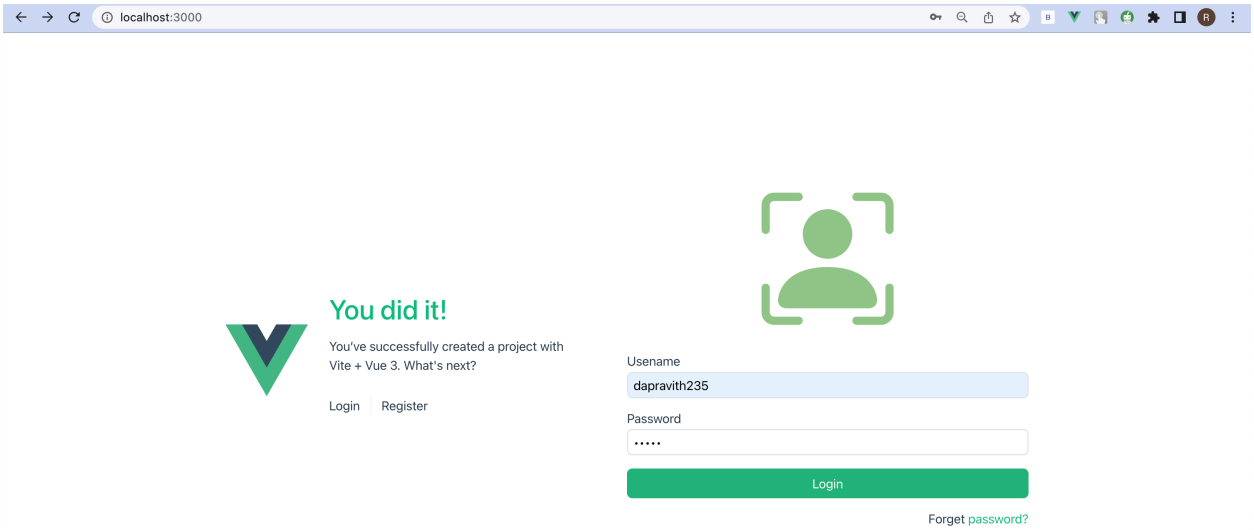
Academic Year: 2022 ~ 2023

Table of Contents

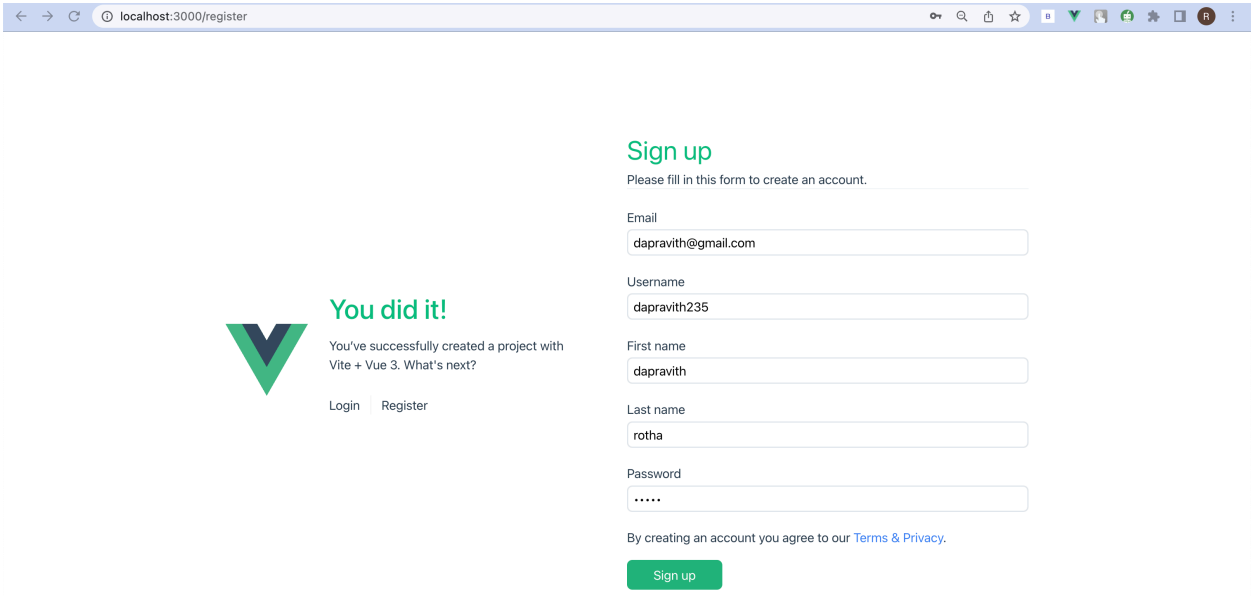
GitHub's Repository: https://github.com/Dapravith/TP_Internet_Programming/tree/TP10

EX1: Integrate the previous authentication APIs with VueJS.

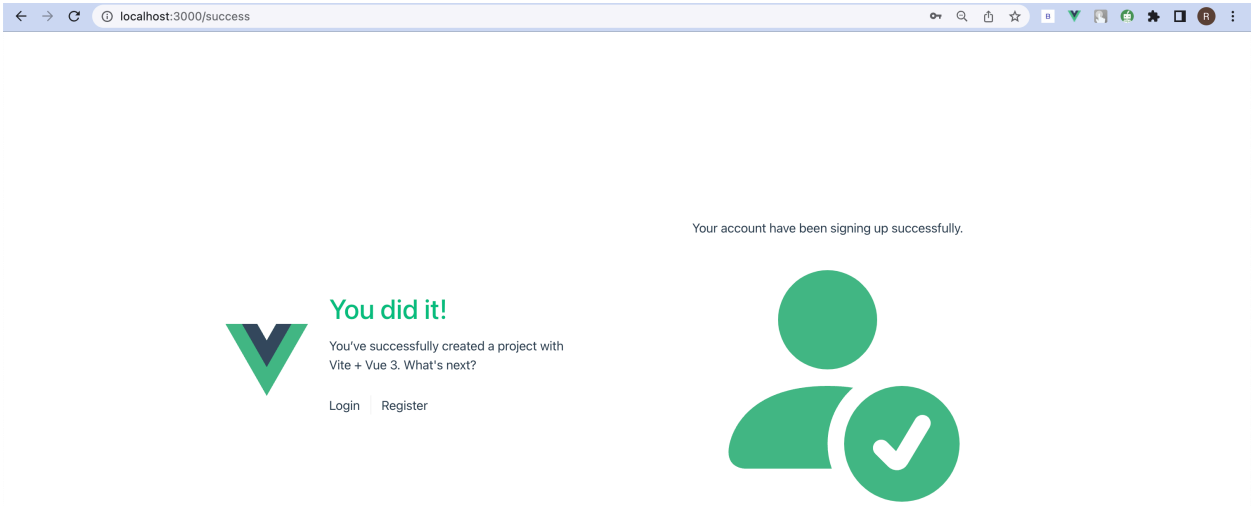
- Result of Exercise1:
- Login



- Register



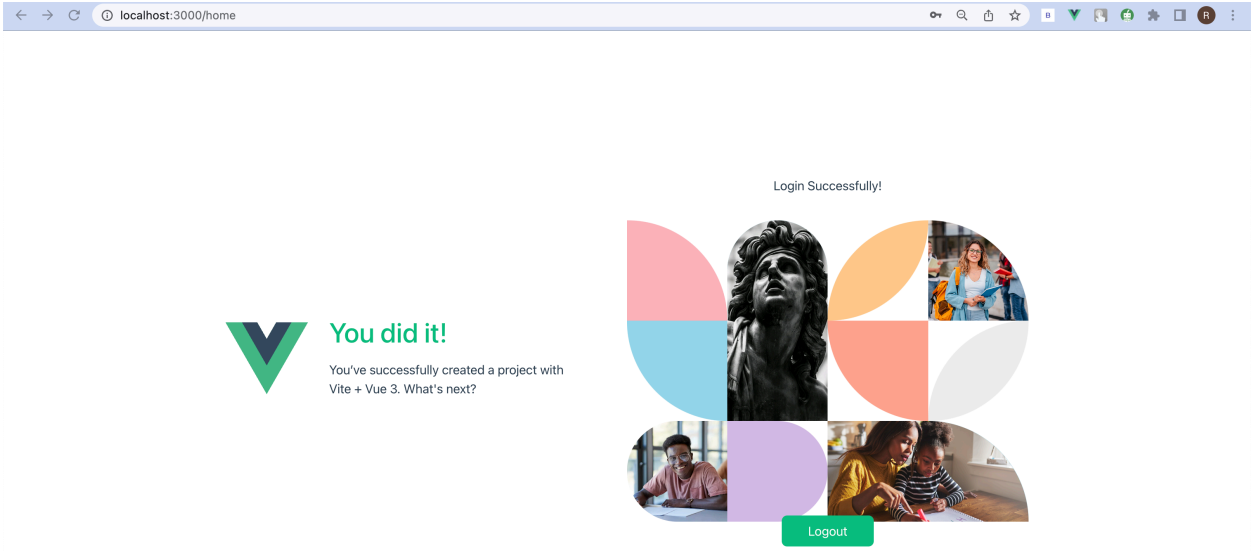
- Register account sucessfully



- Data in MongoDB after register it's load new data in database

Key	Value	Type
(1) 647dff4241e4142c226535f1	{ email : "virak@gmail.com" } (9 fields)	Document
_id	647dff4241e4142c226535f1	ObjectId
username	virak1244	String
updatedAt	6/5/2023, 10:29:06 PM - 8 minutes ago	Date
password	$2a$10$Pj0jpq2ZmvtCaCqiXyAGZeyWKU/MiBz/phUp92OAgoeagolecWc4G	String
lastname	rotha	String
firstname	virak	String
email	virak@gmail.com	String
createdAt	6/5/2023, 10:29:06 PM - 8 minutes ago	Date
__v	0	Int32

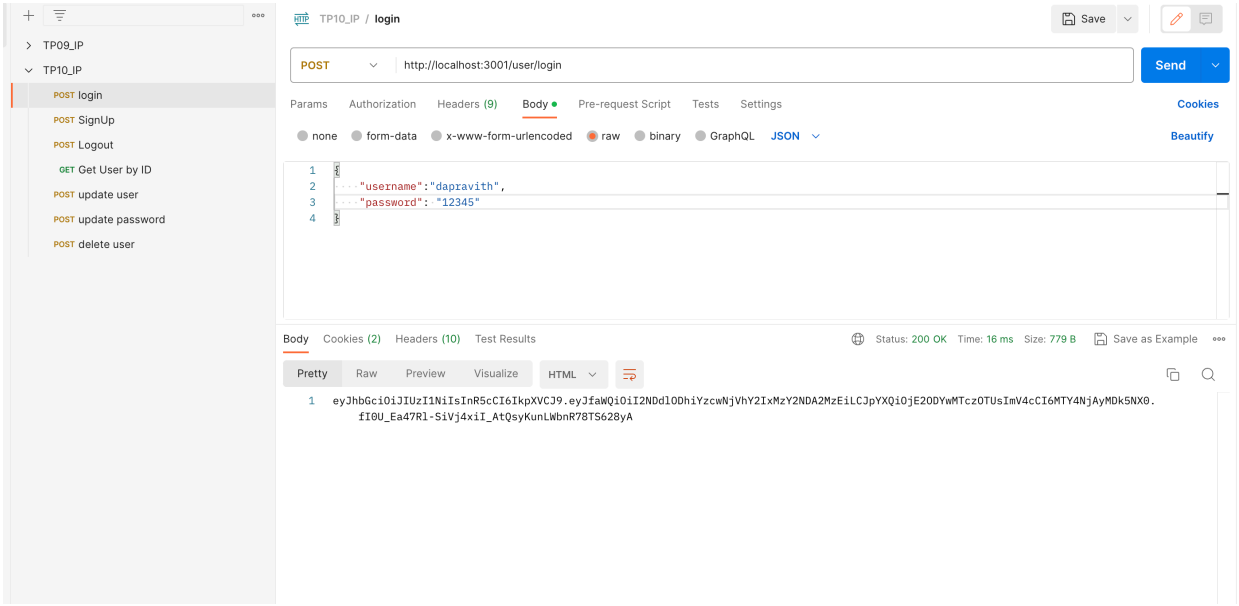
- Home-page



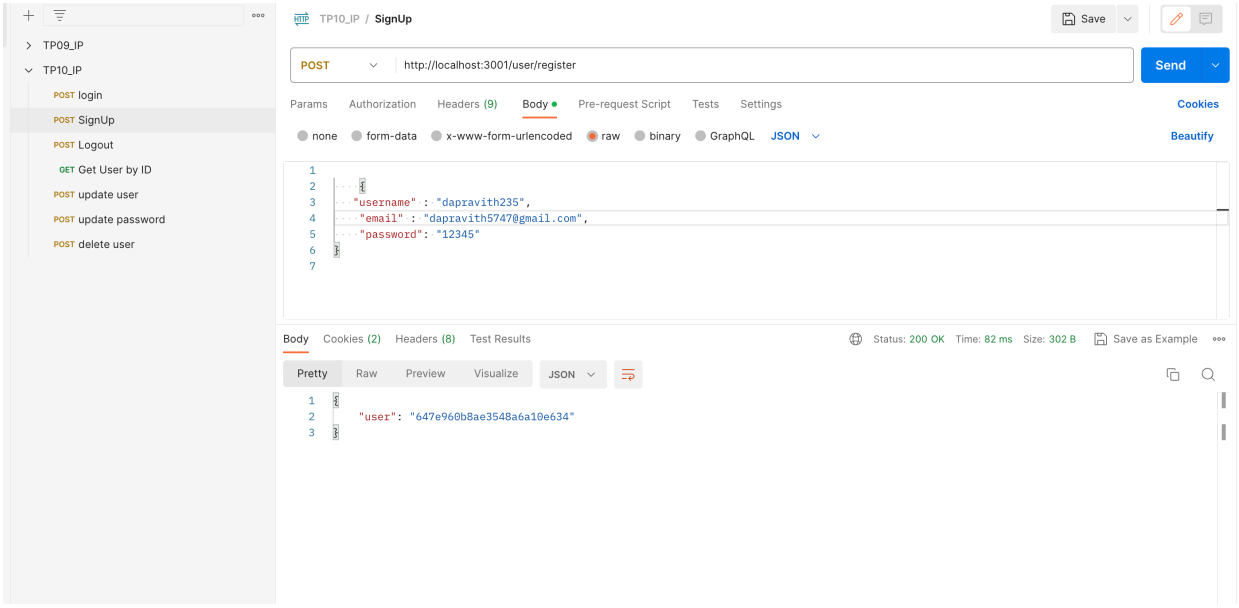
EX2: Continue implementing the authentication and user APIs

- Result of Ex02:

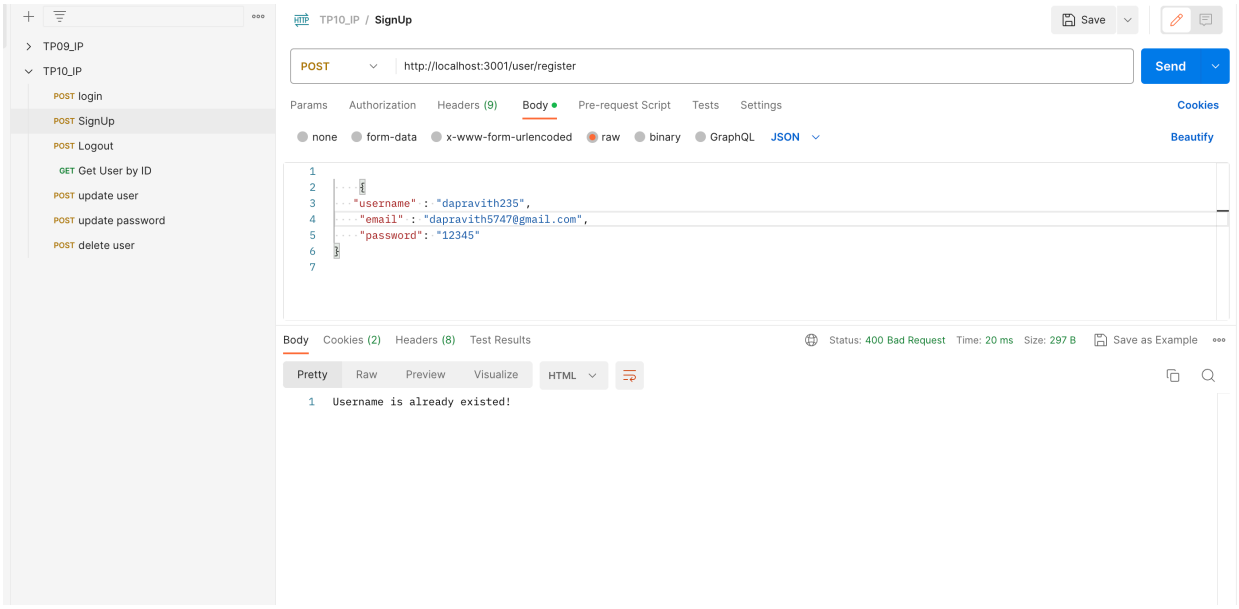
- login user and get token:



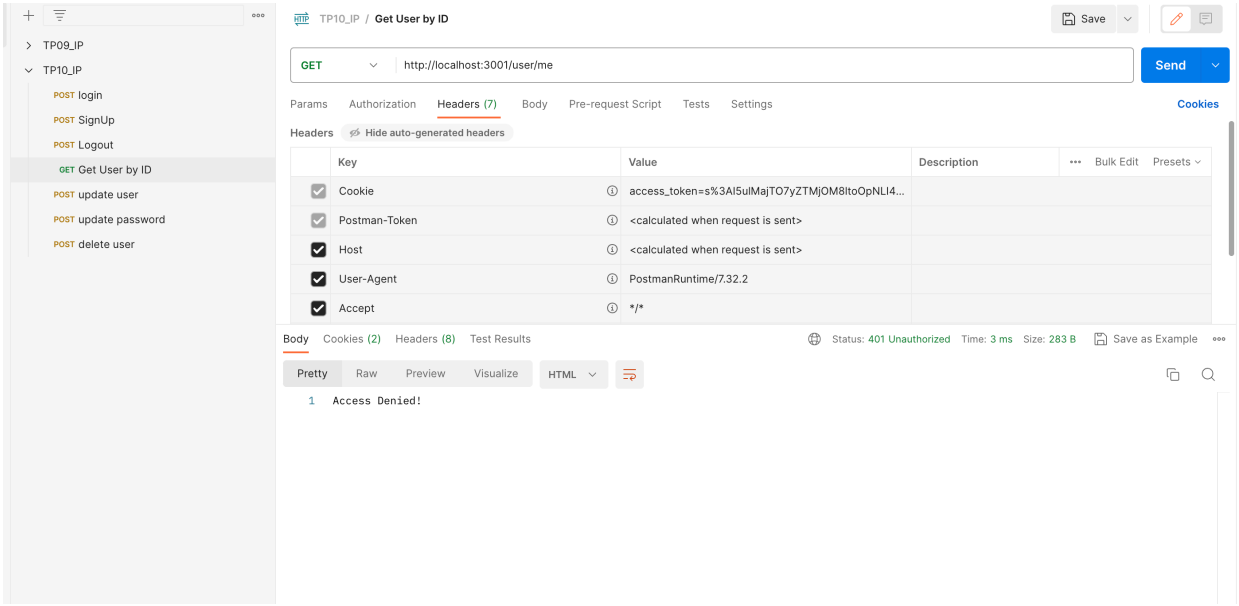
- Register



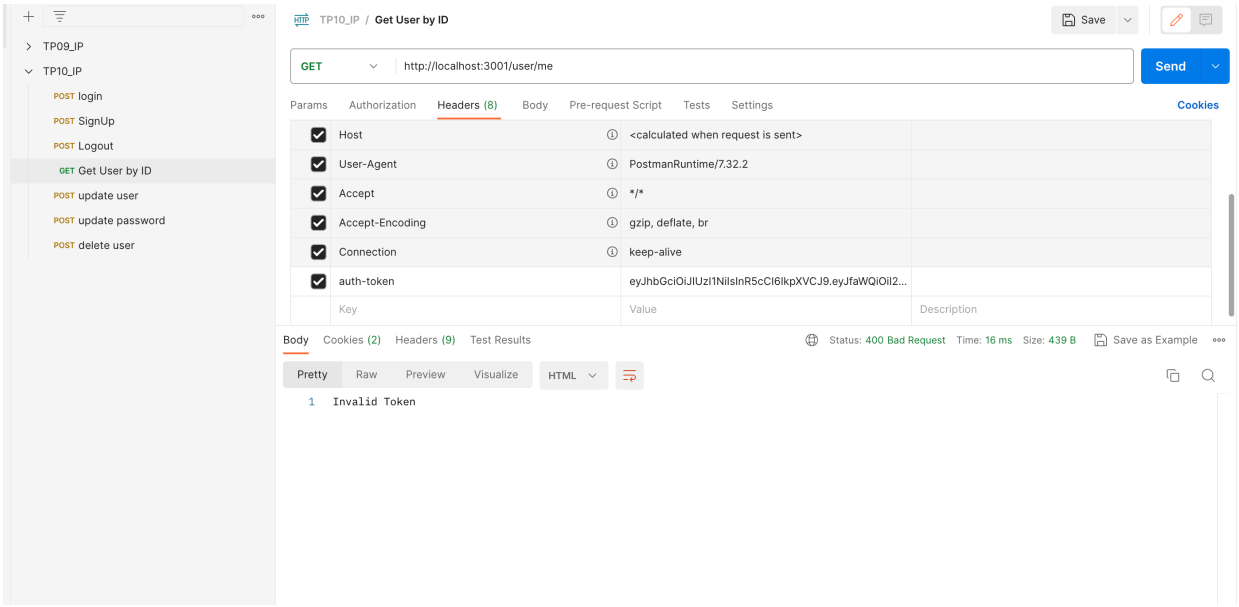
- Register already existed



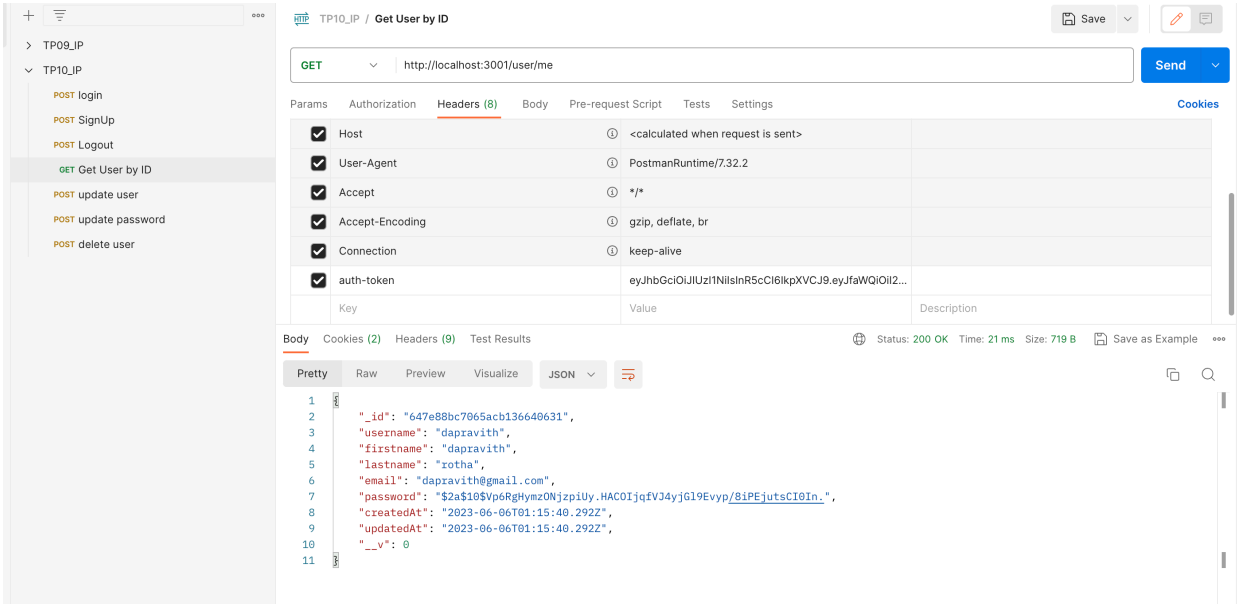
- When no token



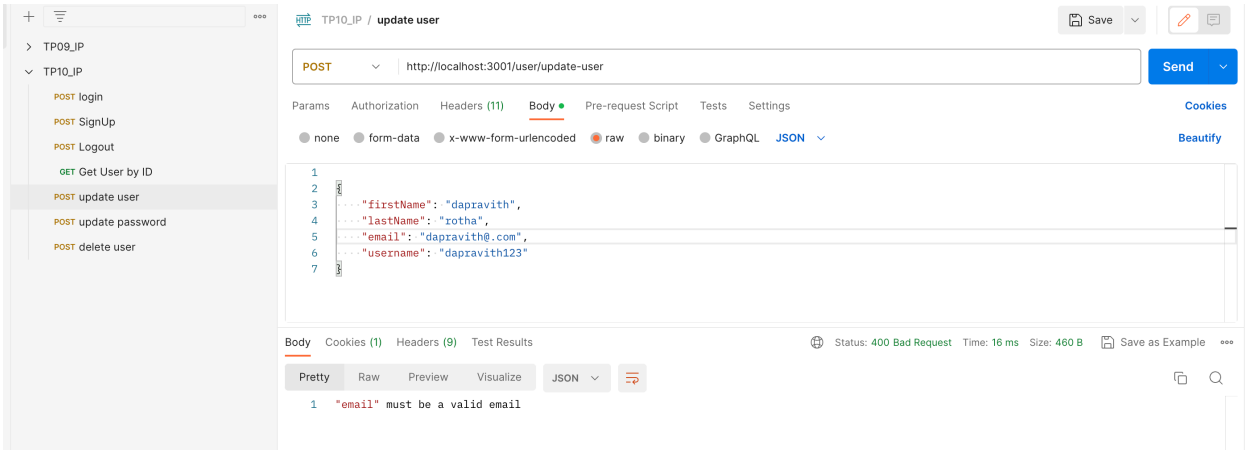
- Invalid or wrong token



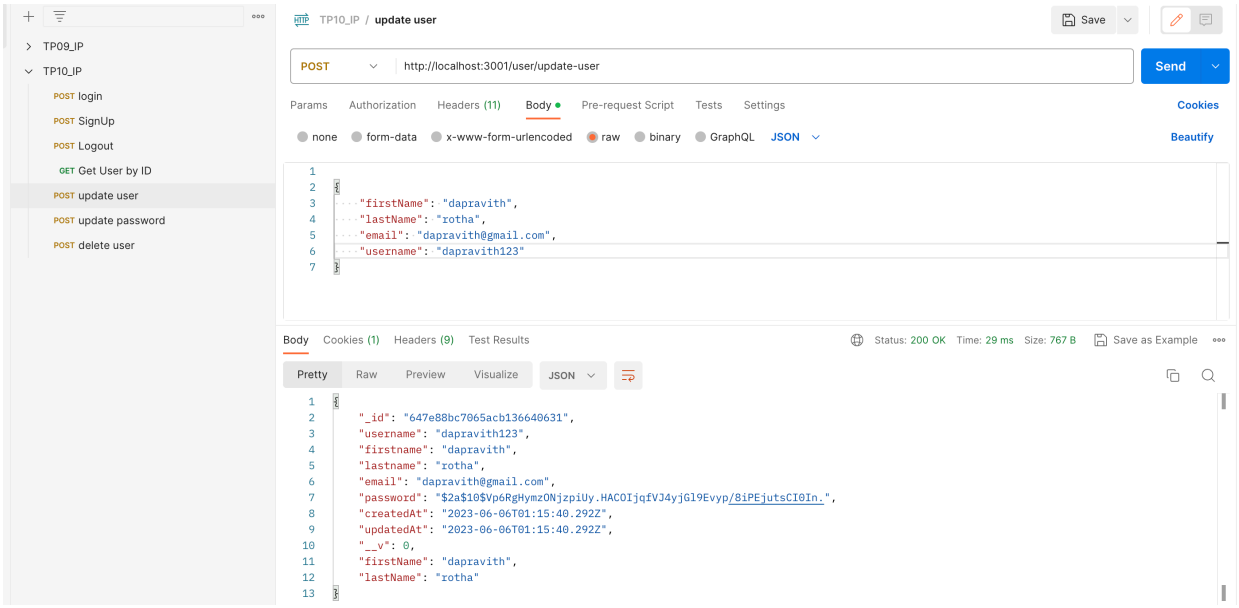
- Access by username via correct token



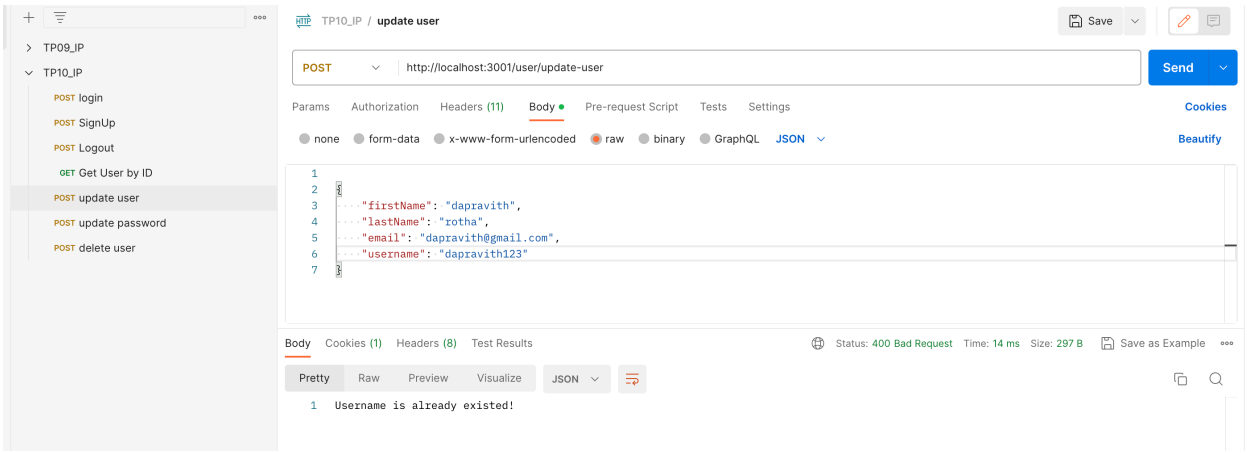
- when update user with invalid email format:



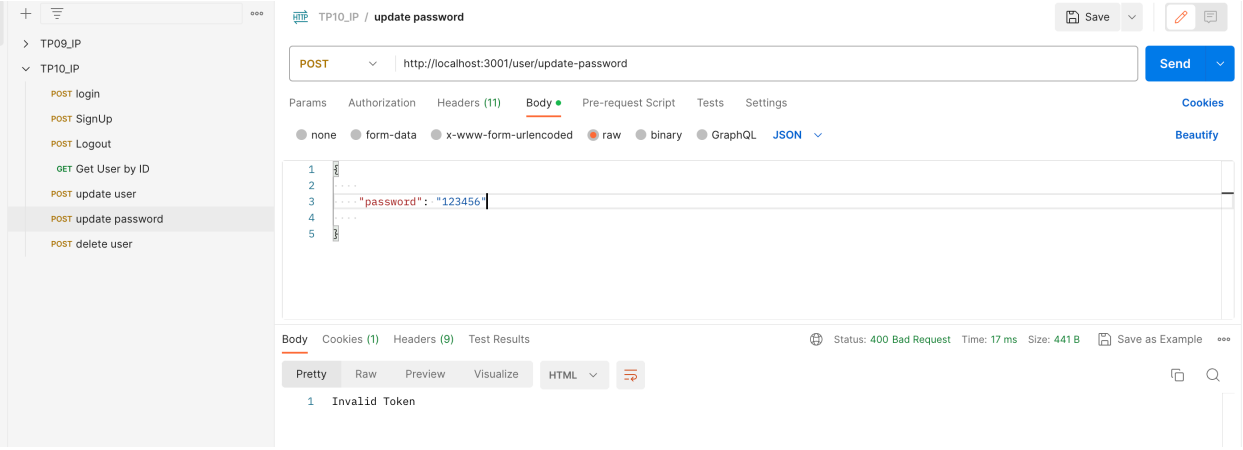
- update user Successful:



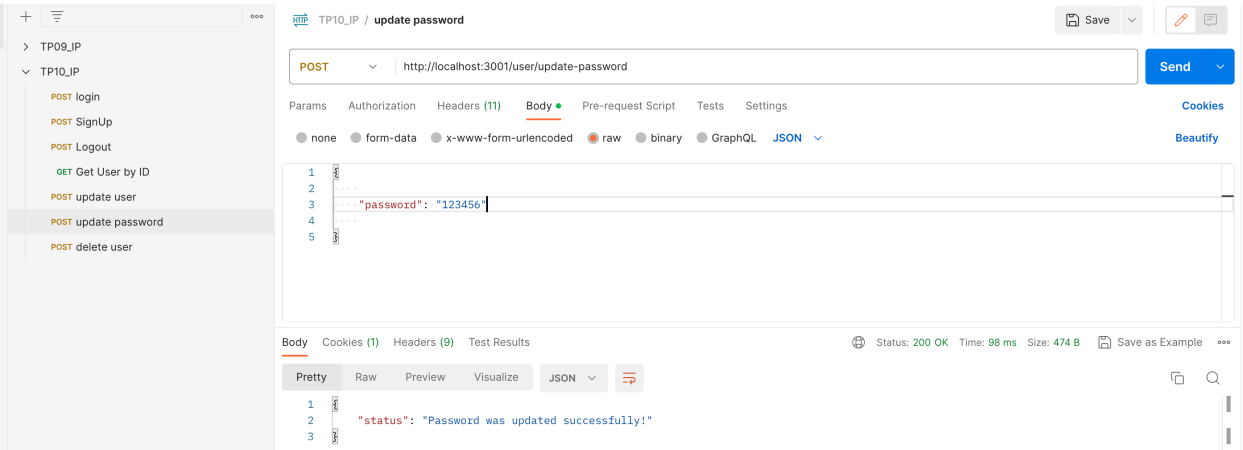
- when update user with exist username account:



- Invalid token for update password:



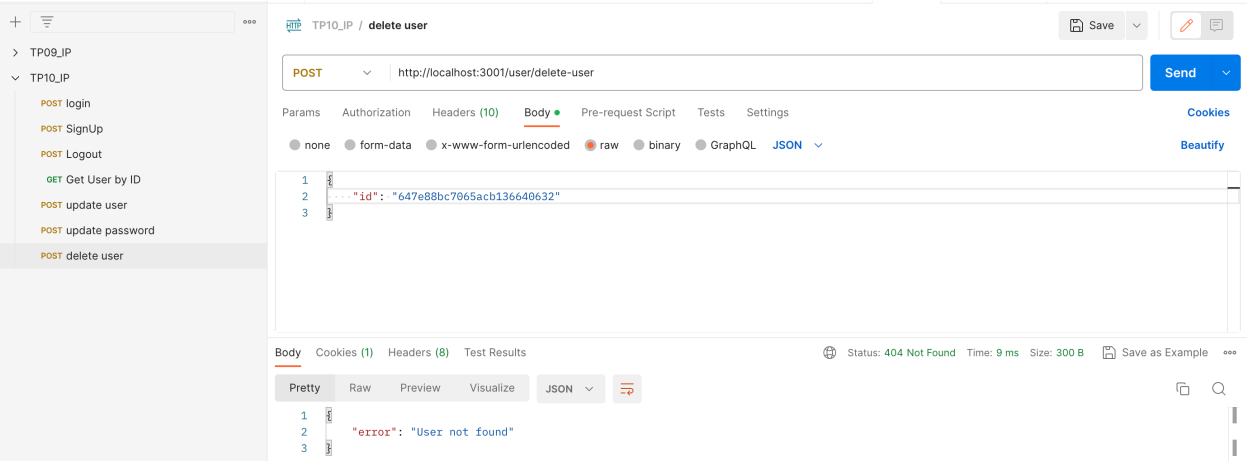
- when update password successful:



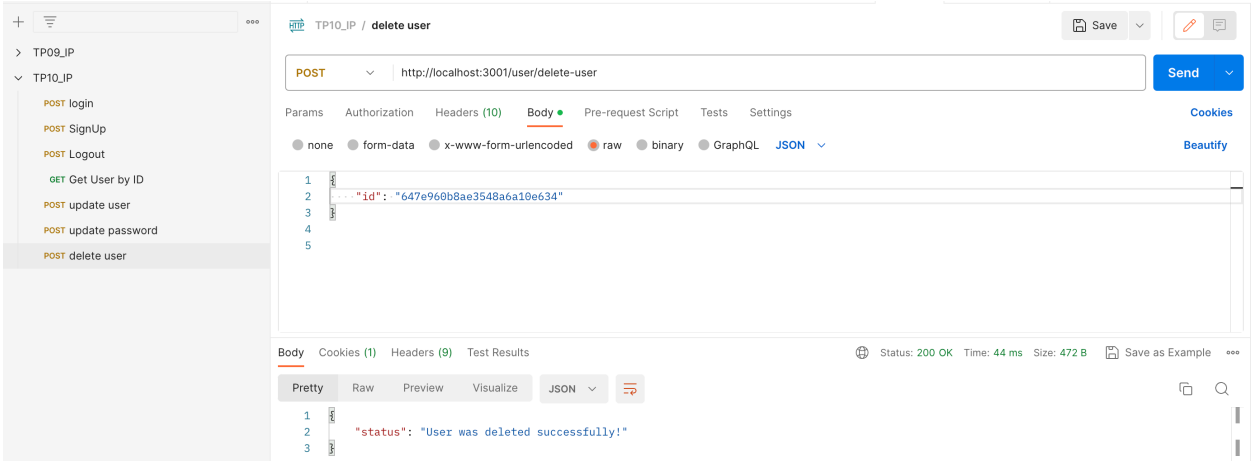
- Delete User
- data before delete:

Key	Value	Type
(1) 647e87266e0d341a5c62c063	{ email : "dapravith@gmail.com" } (9 fields)	Document
_id	647e87266e0d341a5c62c063	ObjectId
username	dapravith	String
updatedAt	6/6/2023, 8:08:54 AM - an hour ago	Date
password	$2a$10$BPJB1QDKoCslnJDxxieOfODkD7mJTBZuWp9PTBeQipnCF6FHMoPRq	String
lastname	rotha	String
firstname	dapravith	String
email	dapravith@gmail.com	String
createdAt	6/6/2023, 8:08:54 AM - an hour ago	Date
__v	0	Int32
(2) 647e86316e0d341a5c62c058	{ email : "virak@gmail.com" } (9 fields)	Document
_id	647e86316e0d341a5c62c058	ObjectId
username	virak124456	String
updatedAt	6/6/2023, 8:04:49 AM - an hour ago	Date
password	$2a$10$AR72mBm68TELsFfthuHWZOaVny/sBkz6SS361mJC0HDPnLnpGKud2	String
lastname	rotha	String
firstname	virak	String
email	virak@gmail.com	String
createdAt	6/6/2023, 8:04:49 AM - an hour ago	Date
__v	0	Int32

- when delete user does not exist in database:



- Delete user successfully.



- data after delete:

Key	Value	Type
(1) 647e93c018ef557bd0fc4c06	{ email : "test1@gmail.com", firstName : "dara", lastName : "vith" } (7 fields)	Document
_id	647e93c018ef557bd0fc4c06	ObjectId
username	test1	String
password	$2b$10$WvTI7gxdsJeCYZq1PHpjXuANEVPWACUxAAx2Cy5k9zmU7pHIA6Ed2	String
lastName	vith	String
firstName	dara	String
email	test1@gmail.com	String
__v	0	Int32
(2) 647dff4241e4142c226535f1	{ email : "virak@gmail.com" } (9 fields)	Document
(3) 647b43f51aa7ca3b36ad45ad	{ email : "dapravithrotha@gmail.com" } (9 fields)	Document
(4) 6472a2836c943499b4b542a8	{ email : "sokdara1234@gmail.com" } (9 fields)	Document
(5) 6472a017aabba63e02e1a7cb	{ email : "chanvotey@gmail.com" } (9 fields)	Document
(6) 6471c90e1ace404c0a8b7ce6	{ email : "SokVatey@gmail.com" } (9 fields)	Document
(7) 6471926a5819ba802ce92d8d	{ email : "channararoth@gmail.com" } (9 fields)	Document
(8) 64718dcf5819ba802ce92d58	{ email : "Guest@gmail.com" } (9 fields)	Document
(9) 64718aa25819ba802ce92d3b	{ email : "admin@gmail.com" } (9 fields)	Document

- data after delete and display the result: