1. Prevent Duplicate Email
   Solusi
   1.1. Untuk mengatasi duplicate email perlu dibuat penambahan beberapa code di repository , service , dan juga controller
   a) Repository

```
/**
* find by email
* @param {string} email - Email
* @returns {Promise}
*/
async function findByEmail(email) {
return User.findOne({ email });
}
```

   b) Service

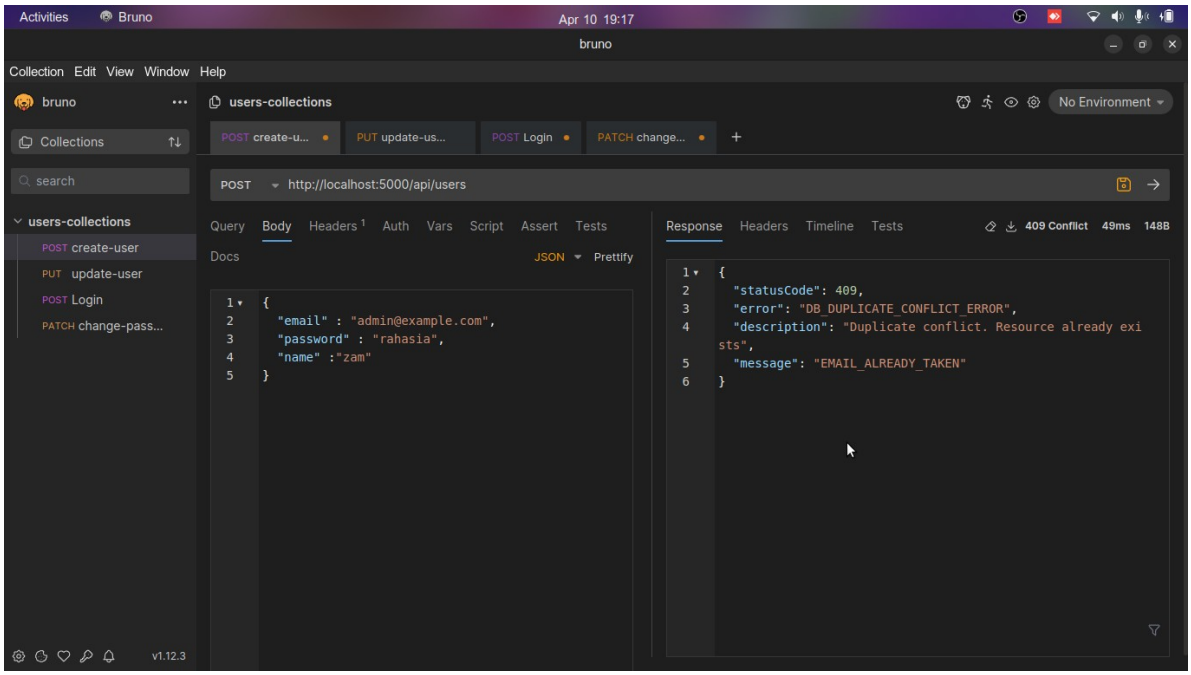
```
async function checkEmail(email) {
var user = await usersRepository.findByEmail(email);
if (user) {
// jika email tidak digunakan return false
return true;
}
return false;
}
```

   c) Controller

```
/**
* Handle create user request
* @param {object} request - Express request object
* @param {object} response - Express response object
* @param {object} next - Express route middlewares
* @returns {object} Response object or pass an error to the next route
*/
async function createUser(request, response, next) {
try {
const name = request.body.name;
const email = request.body.email;
const password = request.body.password;

var emailIsUsed = await usersService.checkEmail(email);
if (emailIsUsed) {
throw errorResponder(
errorTypes.DB_DUPLICATE_CONFLICT,
'EMAIL_ALREADY_TAKEN'
);
}
const success = await usersService.createUser(name, email, password);
if (!success) {
throw errorResponder(
errorTypes.UNPROCESSABLE_ENTITY,
'Failed to create user'
);
}
```

```
return response.status(200).json({ name, email });
} catch (error) {
return next(error);
}
}
```

Bukti



2. Confirm Password
   Solusi
   a) Tambahkan validasi agar confirmPassword dibutuhkan pada userValidator
   createUser: {

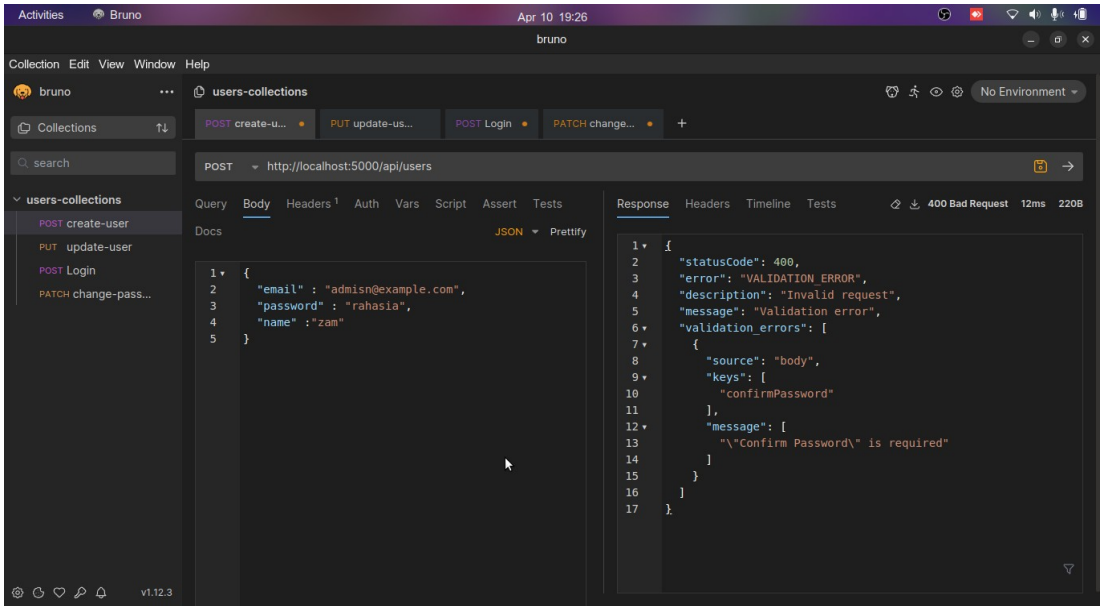
```
body: {
name: joi.string().min(1).max(100).required().label('Name'),
email: joi.string().email().required().label('Email'),
password: joi.string().min(6).max(32).required().label('Password'),
confirmPassword : joi.string().min(6).max(32).required().label('Confirm Password')
},
},
```

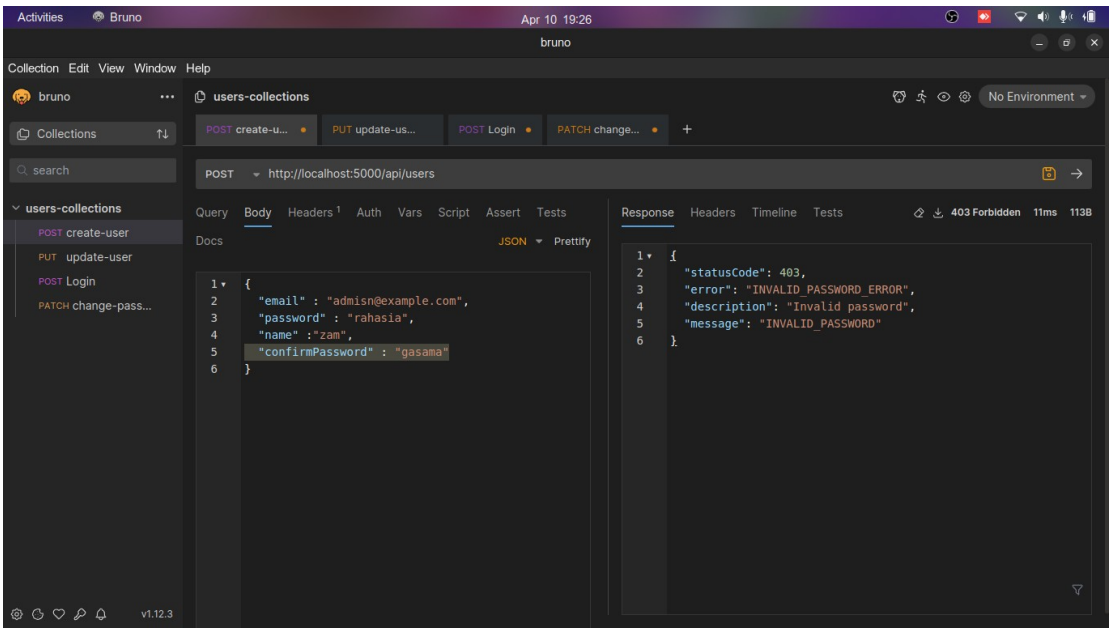
   b) tambahkan kode untuk mengecek pada controller

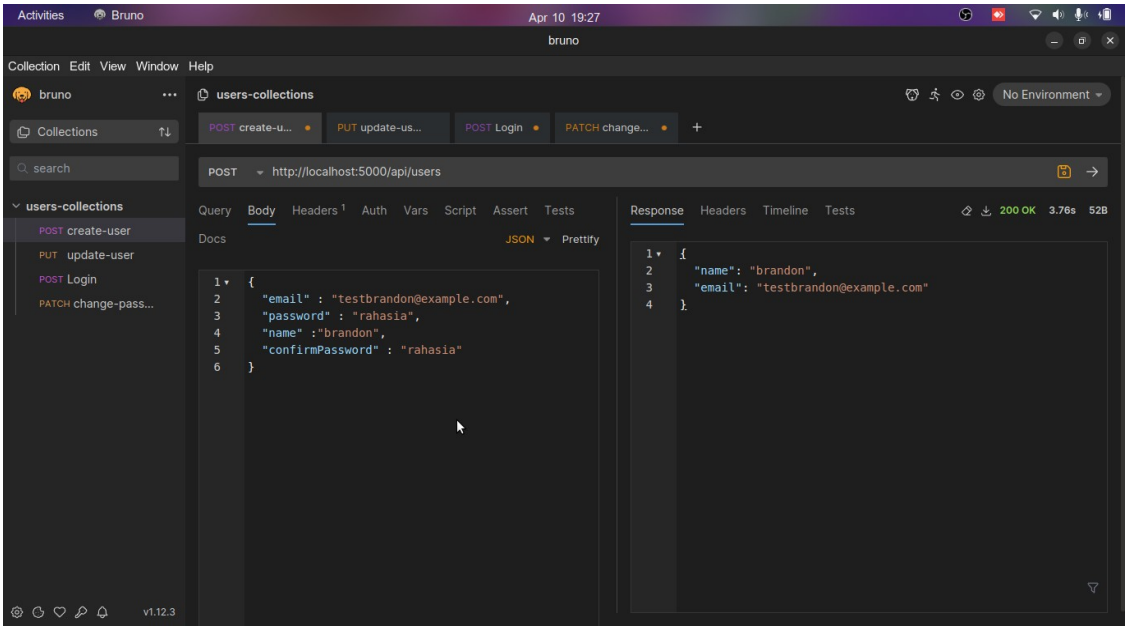
```
async function createUser(request, response, next) {
try {
const name = request.body.name;
const email = request.body.email;
const password = request.body.password;
const confirmPassword = request.body.confirmPassword;

if(password !== confirmPassword){
throw errorResponder(
errorTypes.INVALID_PASSWORD,
'INVALID_PASSWORD'
);
}
```

```
var emailIsUsed = await usersService.checkEmail(email);
if (emailIsUsed) {
throw errorResponder(
errorTypes.DB_DUPLICATE_CONFLICT,
'EMAIL_ALREADY_TAKEN'
);
}
const success = await usersService.createUser(name, email, password);
if (!success) {
throw errorResponder(
errorTypes.UNPROCESSABLE_ENTITY,
'Failed to create user'
);
}

return response.status(200).json({ name, email });
} catch (error) {
return next(error);
}
}
```

Bukti







3. Change Password

Solusi

- Buat route baru

```
route.patch("/:id/change-password" , authenticationMiddleware , celebrate(usersValidator.changePassword) , usersControllers.changePassword)
```

- Buat validator

```
changePassword: {
body: {
oldPassword: joi.string().min(6).max(32).required().label('Old Password'),
newPassword: joi.string().min(6).max(32).required().label('New Password'),
confirmPassword: joi.string().valid(joi.ref('newPassword'))
.required().label('Confirm New Password')
.messages({ 'any.only': '{{#label}} must match new password' }),
```

```
},
},
```

- Buat repository

```
async function changePassword(newPassword , id){
return User.updateOne(
{
 _id: id,
},
{
$set: {
password : newPassword,
},
}
);
}
```

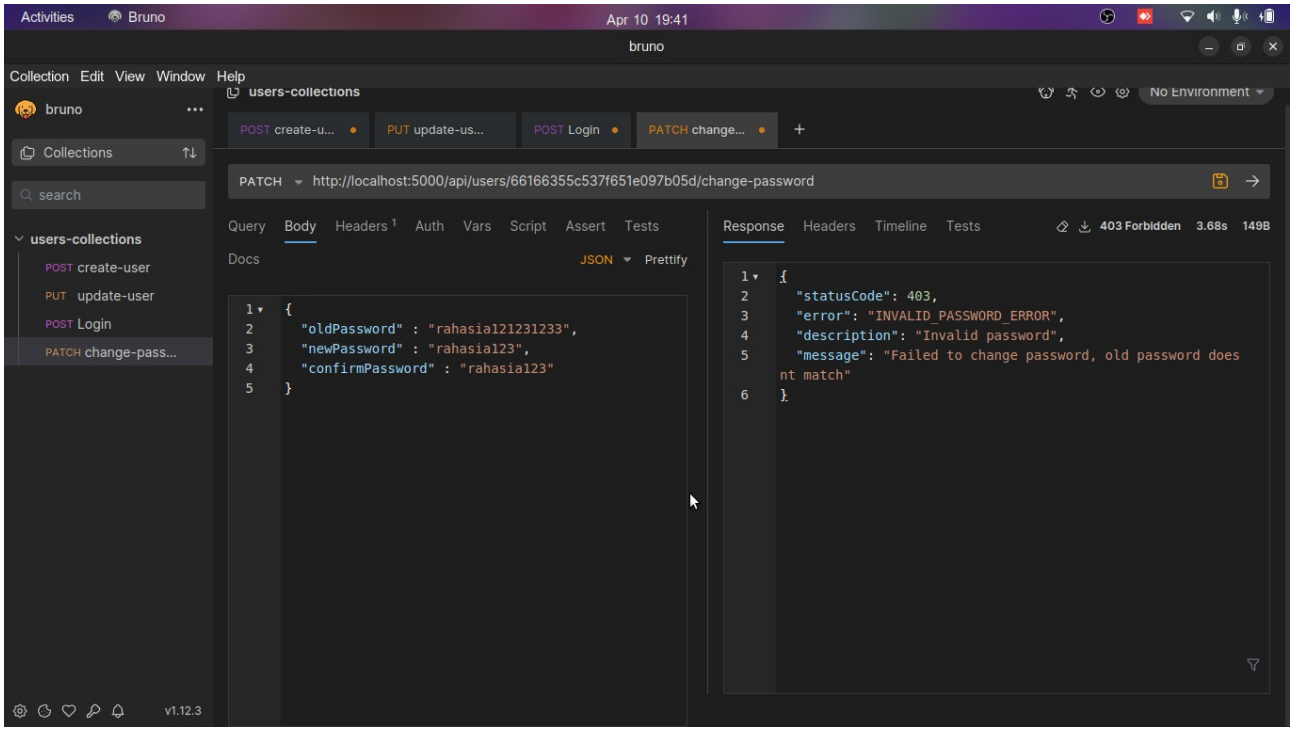
- Buat Service

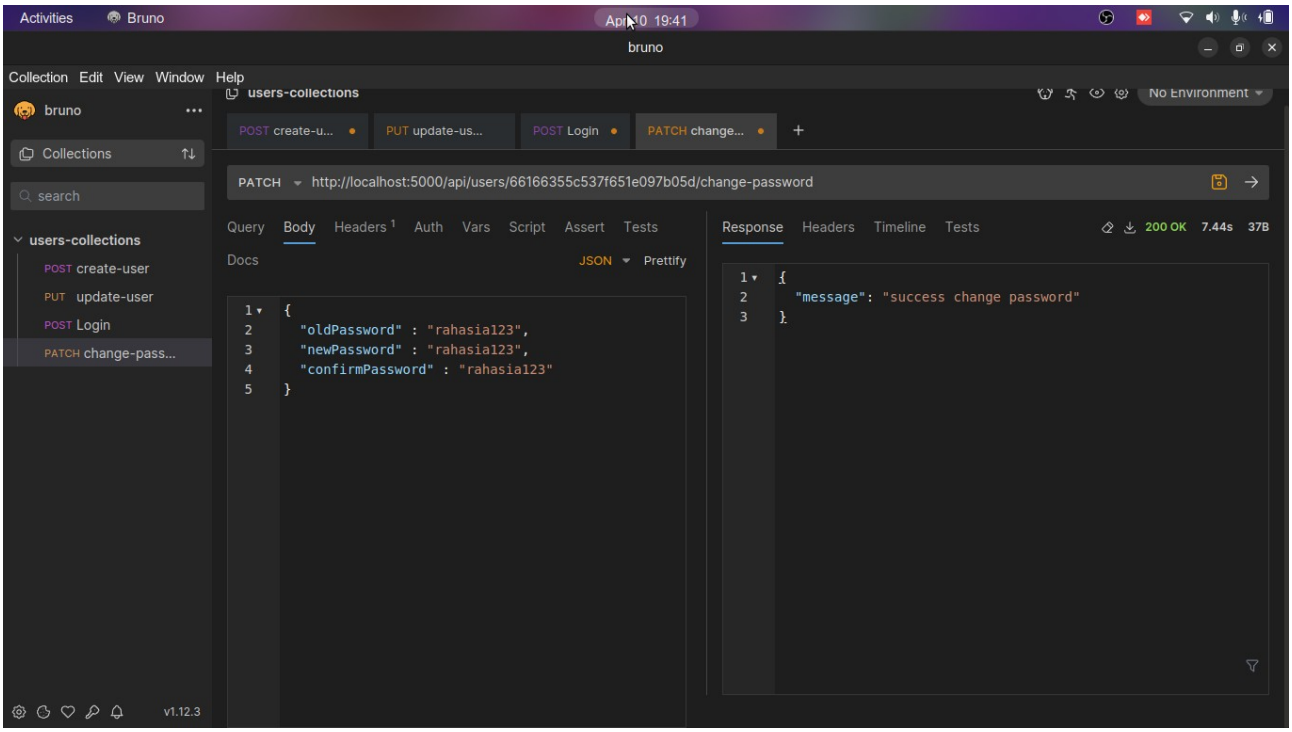
```
async function changePassword(newPassword, oldPassword, id) {
try {
const user = await usersRepository.getUser(id);
const userPassword = user ? user.password : '<RANDOM_PASSWORD_FILLER>';
const passwordChecked = await passwordMatched(oldPassword, userPassword);
if (user && passwordChecked) {
let hashedPassword = await hashPassword(newPassword);
return usersRepository.changePassword(hashedPassword , id)
} else {
return null;
}
} catch (error) {
return null;
}
}
```

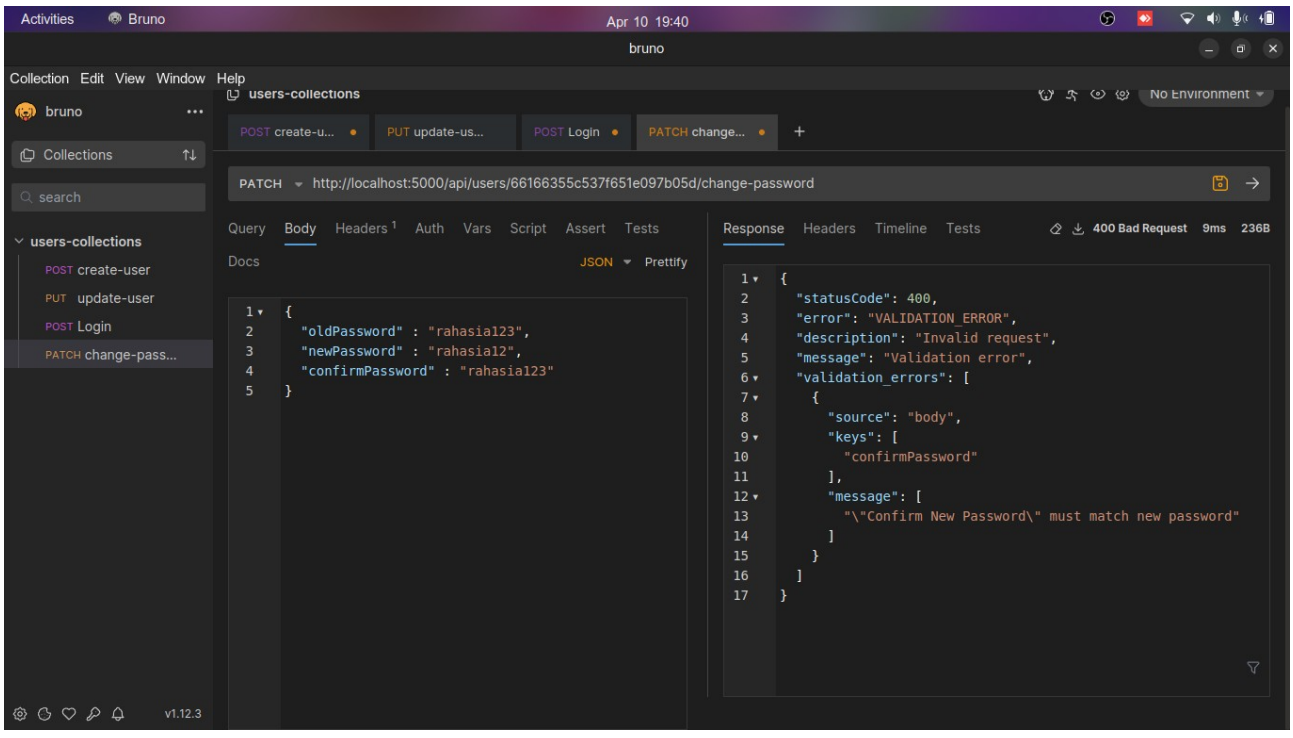
- Buat controller

```
async function changePassword(request, response, next) {
try {
const id = request.params.id;
const newPassword = request.body.newPassword;
const oldPassword = request.body.oldPassword;
let user = await usersService.changePassword(newPassword, oldPassword, id);
if (!user) {
throw errorResponder(
errorTypes.INVALID_PASSWORD,
'Failed to change password, old password doesnt match'
);
}
return response.status(200).json({ message: 'success change password' });
} catch (error) {
return next(error);
}
}
```

Bukti


Activities
Bruno
Apr 10 19:41
bruno
Collection Edit View Window Help
bruno
Collections
search
users-collections
POST create-user
PUT update-user
POST Login
PATCH change-pass...
users-collections
No Environment
POST create-u...
PUT update-us...
POST Login
PATCH change...
PATCH http://localhost:5000/api/users/66166355c537f651e097b05d/change-password
Query Body Headers 1 Auth Vars Script Assert Tests
Docs
JSON Prettify
{
"oldPassword" : "rahasia121231233",
"newPassword" : "rahasia123",
"confirmPassword" : "rahasia123"
}
Response Headers Timeline Tests
403 Forbidden 3.68s 149B
{
"statusCode": 403,
"error": "INVALID_PASSWORD_ERROR",
"description": "Invalid password",
"message": "Failed to change password, old password does nt match"
}
v1.12.3


Activities
Bruno
Apr 10 19:41
bruno
Collection Edit View Window Help
bruno
Collections
search
users-collections
POST create-user
PUT update-user
POST Login
PATCH change-pass...
v1.12.3
users-collections
No Environment
POST create-u...
PUT update-us...
POST Login
PATCH change...
PATCH http://localhost:5000/api/users/66166355c537f651e097b05d/change-password
Query Body Headers 1 Auth Vars Script Assert Tests
Docs
JSON Prettify
{
"oldPassword" : "rahasia123",
"newPassword" : "rahasia123",
"confirmPassword" : "rahasia123"
}
Response Headers Timeline Tests
200 OK 7.44s 37B
{
"message": "success change password"
}


Activities
Bruno
Apr 10 19:40
bruno
Collection Edit View Window Help
bruno
Collections
search
users-collections
POST create-user
PUT update-user
POST Login
PATCH change-pass...
v1.12.3
users-collections
No Environment
POST create-u...
PUT update-us...
POST Login
PATCH change...
PATCH http://localhost:5000/api/users/66166355c537f651e097b05d/change-password
Query Body Headers 1 Auth Vars Script Assert Tests
Docs
JSON Prettify
{
"oldPassword" : "rahasia123",
"newPassword" : "rahasia12",
"confirmPassword" : "rahasia123"
}
Response Headers Timeline Tests
400 Bad Request 9ms 236B
{
"statusCode": 400,
"error": "VALIDATION_ERROR",
"description": "Invalid request",
"message": "Validation error",
"validation_errors": [
{
"source": "body",
"keys": [
"confirmPassword"
],
"message": [
"\"Confirm New Password\" must match new password"
]
}
]
}