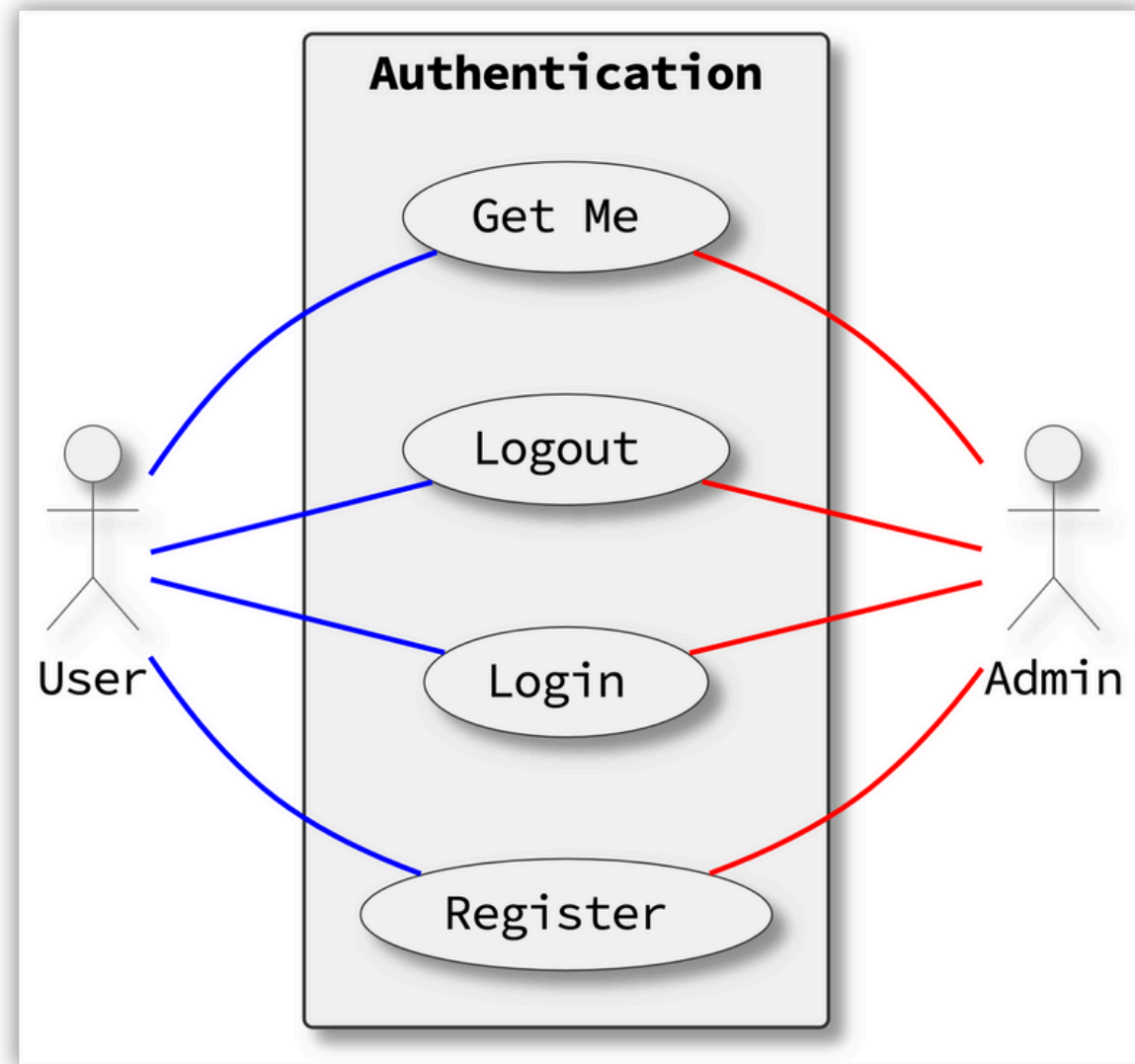# Final Backend Project

2110503 Software Development Practice (2024/2) - CEDT

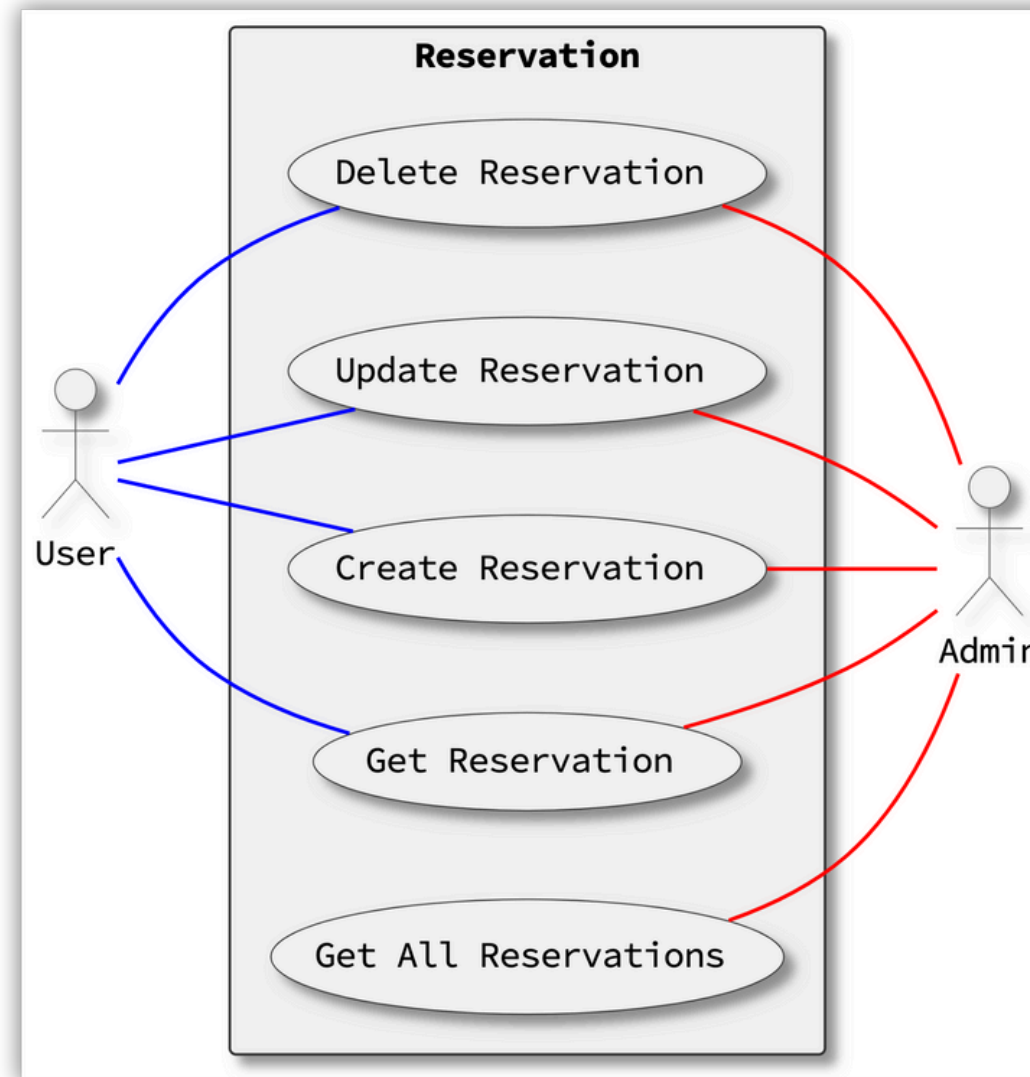**SaeMaGuTae Group**

👤 6733013521 Kritsada Limsripraphan   👤 6733013521 Khannapong Wangkrasaer
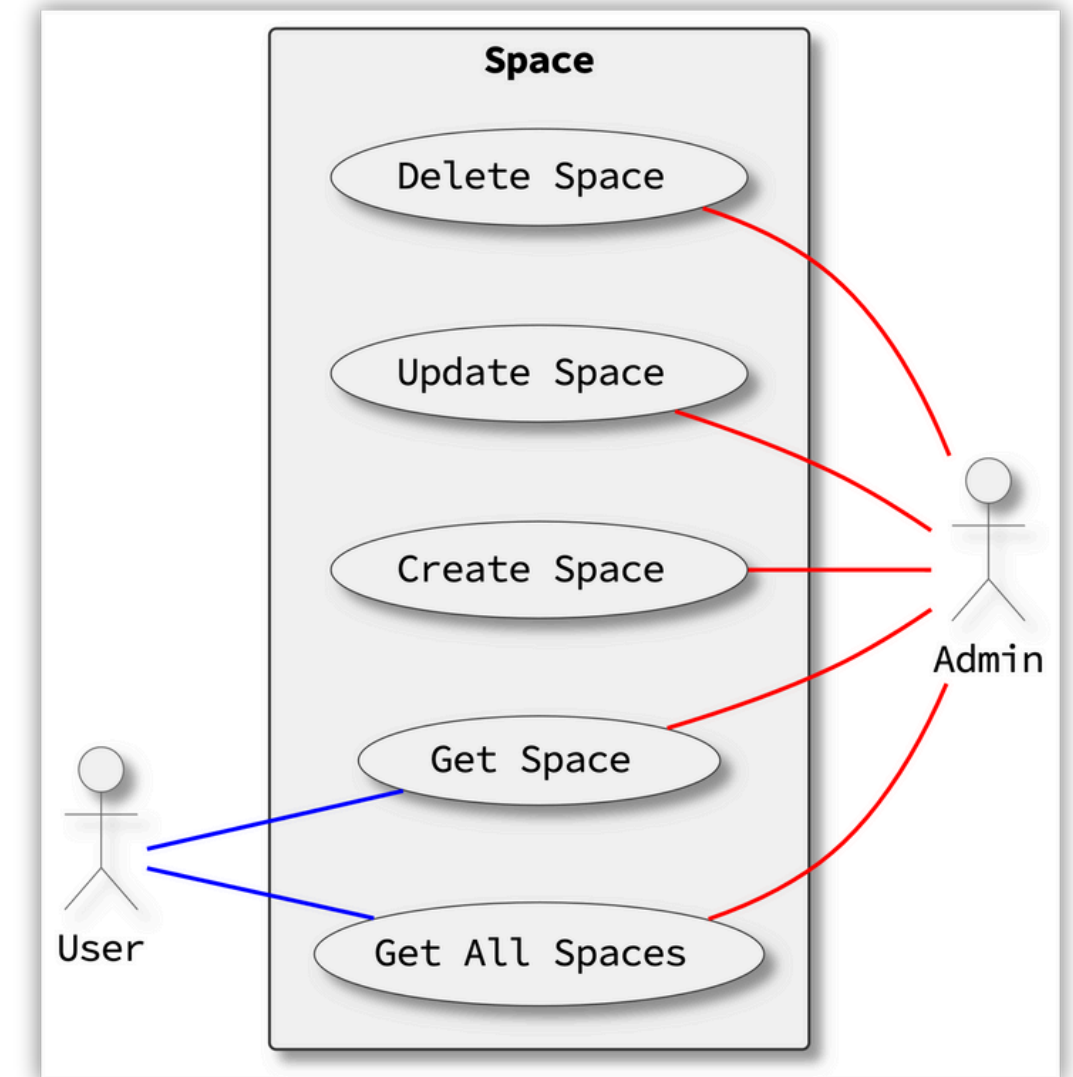
Use Case Diagram

Authentication

Reservation

Space

Class Diagram

AuthController --> User   #Red
AuthController --> AuthMiddleware   #Red

SpaceController --> Space   #Green
SpaceController --> AuthMiddleware   #Green

ReservationController --> AuthMiddleware #Blue
ReservationController --> Reservation   #Blue
ReservationController --> Space   #Blue
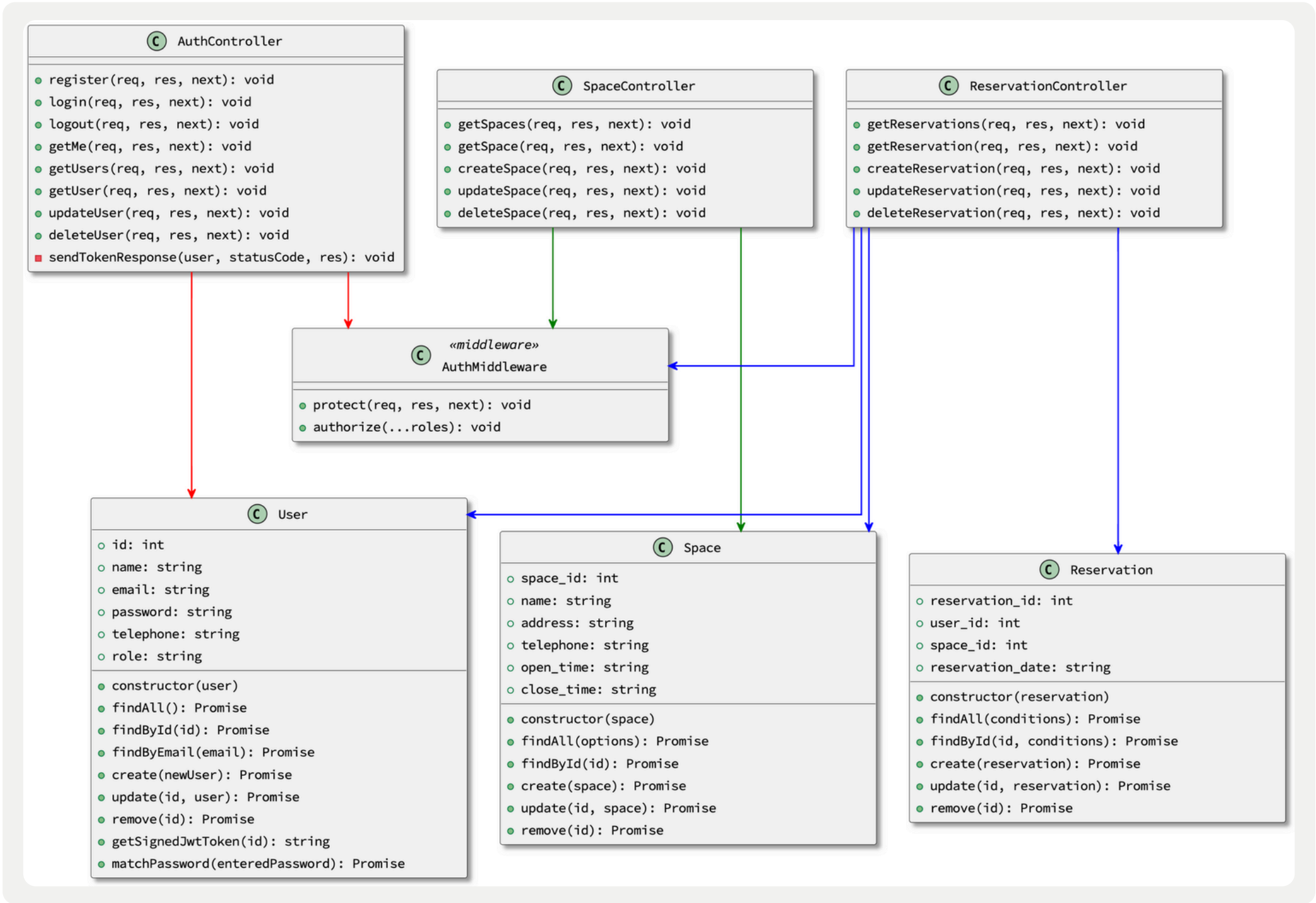ReservationController --> User   #Blue

link to photo

AuthController
- register(req, res, next): void
- login(req, res, next): void
- logout(req, res, next): void
- getMe(req, res, next): void
- getUsers(req, res, next): void
- getUser(req, res, next): void
- updateUser(req, res, next): void
- deleteUser(req, res, next): void
- sendTokenResponse(user, statusCode, res): void

SpaceController
- getSpaces(req, res, next): void
- getSpace(req, res, next): void
- createSpace(req, res, next): void
- updateSpace(req, res, next): void
- deleteSpace(req, res, next): void

ReservationController
- getReservations(req, res, next): void
- getReservation(req, res, next): void
- createReservation(req, res, next): void
- updateReservation(req, res, next): void
- deleteReservation(req, res, next): void

«middleware» AuthMiddleware
- protect(req, res, next): void
- authorize(...roles): void

User
- id: int
- name: string
- email: string
- password: string
- telephone: string
- role: string

- constructor(user)
- findAll(): Promise
- findById(id): Promise
- findByEmail(email): Promise
- create(newUser): Promise
- update(id, user): Promise
- remove(id): Promise
- getSignedJwtToken(id): string
- matchPassword(enteredPassword): Promise

Space
- space_id: int
- name: string
- address: string
- telephone: string
- open_time: string
- close_time: string

- constructor(space)
- findAll(options): Promise
- findById(id): Promise
- create(space): Promise
- update(id, space): Promise
- remove(id): Promise

Reservation
- reservation_id: int
- user_id: int
- space_id: int
- reservation_date: string

- constructor(reservation)
- findAll(conditions): Promise
- findById(id, conditions): Promise
- create(reservation): Promise
- update(id, reservation): Promise
- remove(id): Promise
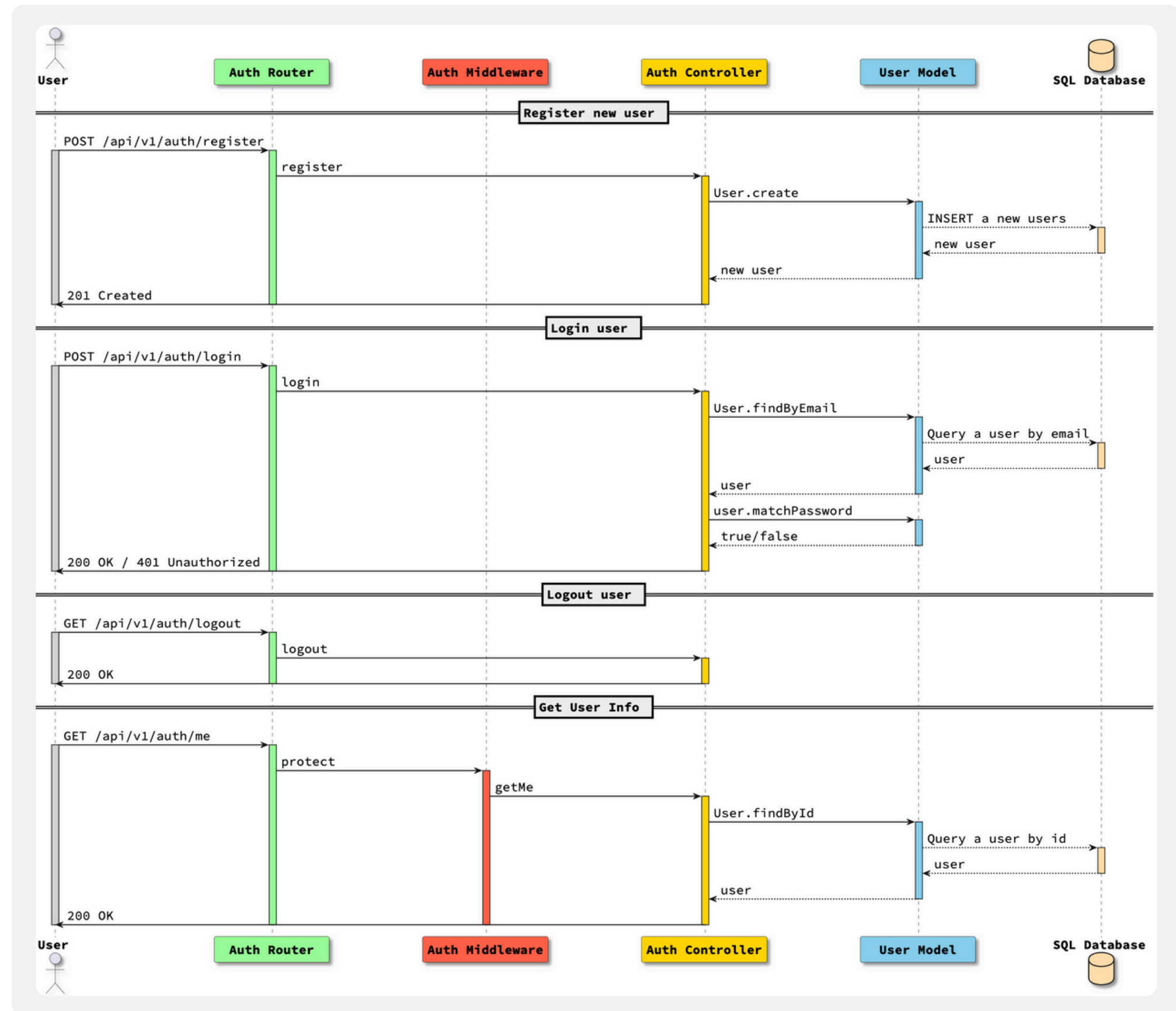
3

Sequence Diagram

```
router.post('/register', register);
router.post('/login', login);
router.get('/logout', logout);
router.get('/me', protect, getMe);
```
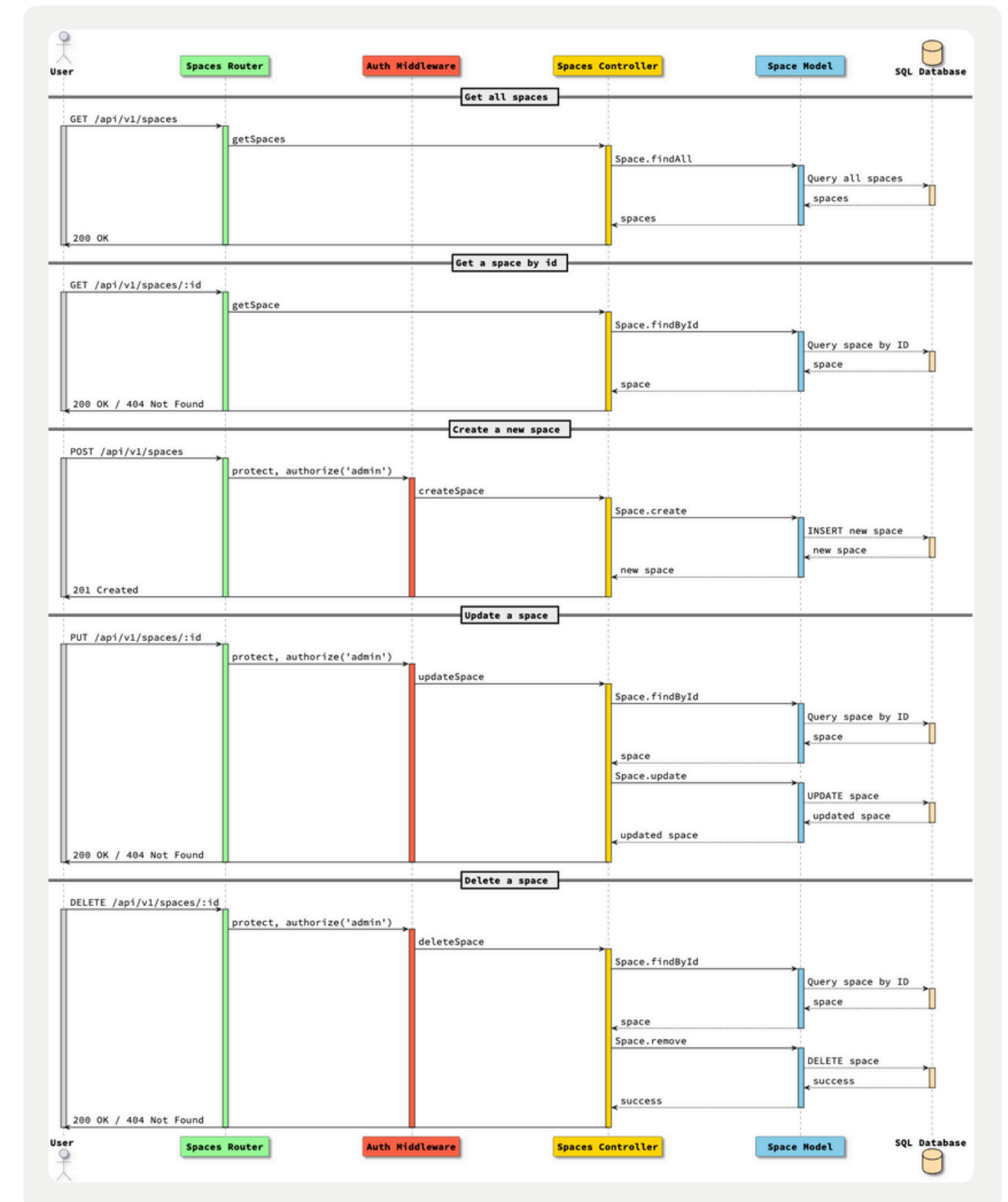
[link to photo](#)

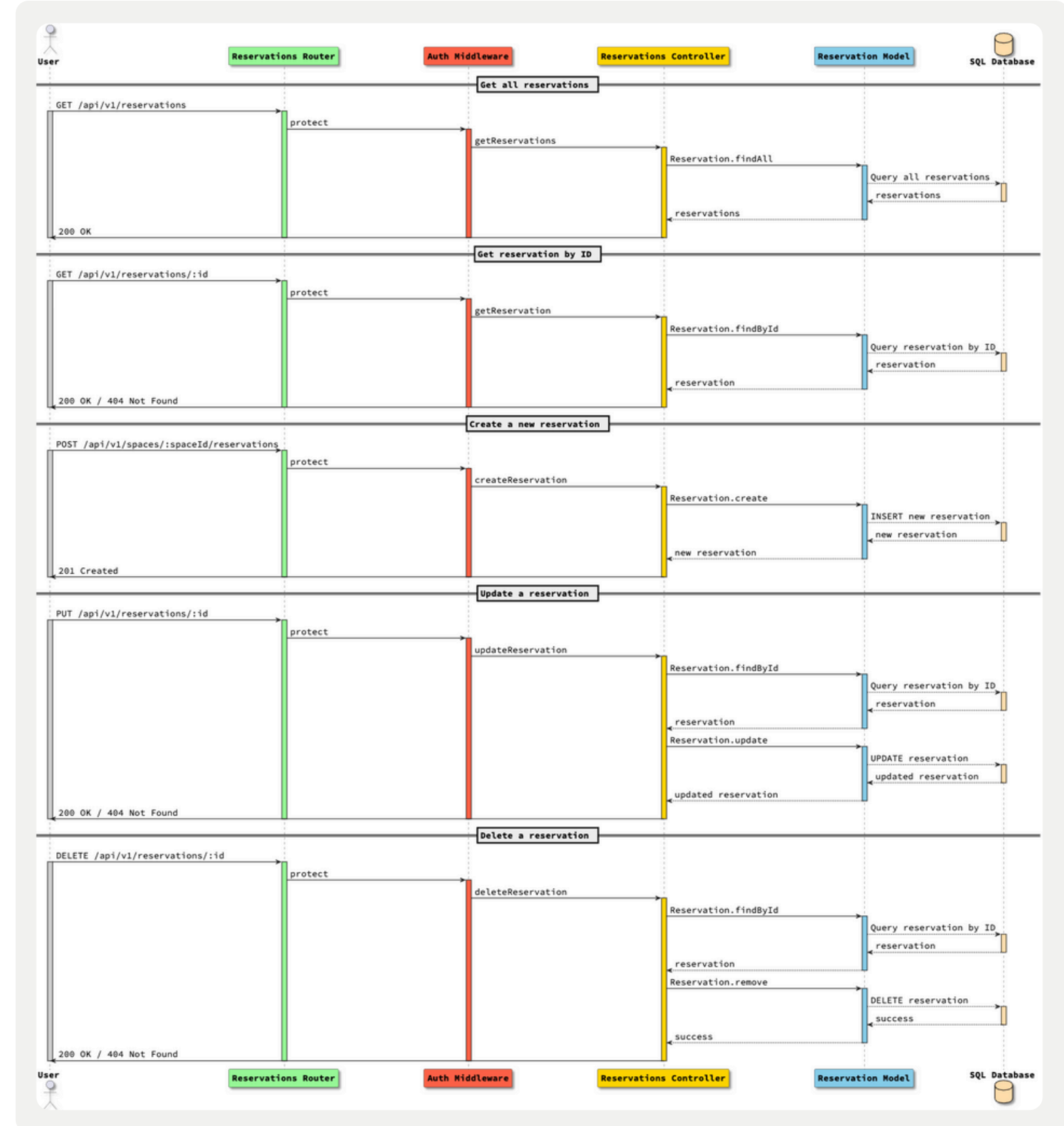4

Sequence Diagram

```
router.route('/')
    .get(getSpaces)
    .post(protect, authorize('admin'), createSpace);

router.route('/:id')
    .get(getSpace)
    .put(protect, authorize('admin'), updateSpace)
    .delete(protect, authorize('admin'), deleteSpace);
```

link to photo

Sequence Diagram

```
router.route('/')
    .get(protect, getReservations)
    .post(protect, createReservation);

router.route('/:id')
    .get(protect, getReservation)
    .put(protect, updateReservation)
    .delete(protect, deleteReservation);
```

link to photo

Activity - Github Activity

- Authentication part
- Co-working Space part
- Reservation part

- documents
- testcase
- extra credits - link

https://github.com/2110503-2564/project-saemagutae