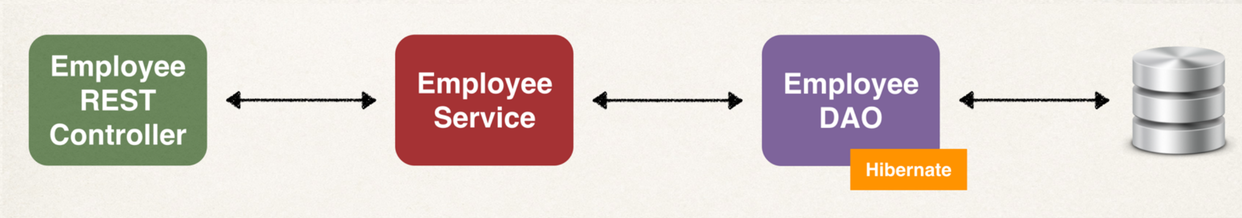
Rest Controller Methods - Update and Delete

Development Process

Step-By-Step

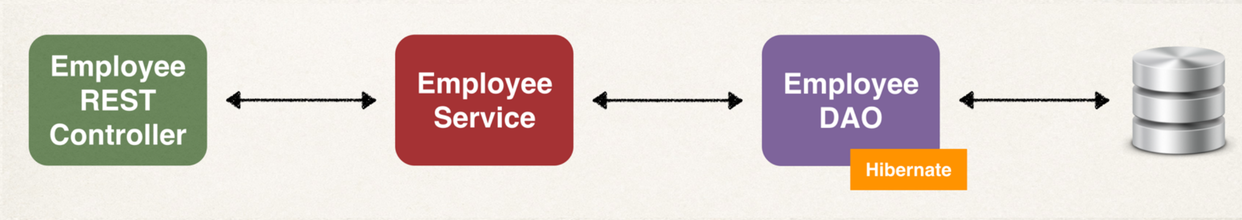
1. Set up Database Dev Environment
2. Create Spring Boot project using Spring Initializr
3. Get list of employees
4. Get single employee by ID
5. Add a new employee
6. Update an existing employee
7. Delete an existing employee



Development Process

Step-By-Step

1. Set up Database Dev Environment
2. Create Spring Boot project using Spring Initializr
3. Get list of employees
4. Get single employee by ID
5. Add a new employee
6. Update an existing employee
7. Delete an existing employee

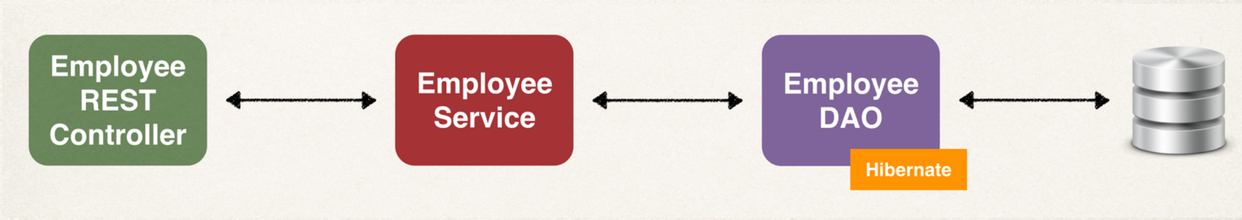


Development Process

Step-By-Step

1. Set up Database Dev Environment
2. Create Spring Boot project using Spring Initializr
3. Get list of employees
4. Get single employee by ID
5. Add a new employee
6. Update an existing employee
7. Delete an existing employee

Rest Controller methods



Real-Time Project

HTTP Method		CRUD Action
GET	/api/employees	Read a list of employees
GET	/api/employees/{employeeId}	Read a single employee
POST	/api/employees	Create a new employee
PUT	/api/employees	Update an existing employee
DELETE	/api/employees/{employeeId}	Delete an existing employee

Real-Time Project

Checkpoint

HTTP Method		CRUD Action
GET	/api/employees	Read a list of employees
GET	/api/employees/{employeeId}	Read a single employee
POST	/api/employees	Create a new employee
PUT	/api/employees	Update an existing employee
DELETE	/api/employees/{employeeId}	Delete an existing employee

Real-Time Project

Checkpoint

HTTP Method		CRUD Action
GET	/api/employees	Read a list of employees
GET	/api/employees/{employeeId}	Read a single employee
POST	/api/employees	Create a new employee
PUT	/api/employees	Update an existing employee
DELETE	/api/employees/{employeeId}	Delete an existing employee

Real-Time Project

HTTP Method		CRUD Action
GET	/api/employees	Read a list of employees
GET	/api/employees/{employeeId}	Read a single employee
POST	/api/employees	Create a new employee
PUT	/api/employees	Update an existing employee
DELETE	/api/employees/{employeeId}	Delete an existing employee

Real-Time Project

HTTP Method		CRUD Action
GET	/api/employees	Read a list of employees
GET	/api/employees/{employeeId}	Read a single employee
POST	/api/employees	Create a new employee
PUT	/api/employees	Update an existing employee
DELETE	/api/employees/{employeeId}	Delete an existing employee

Update Employee

REST Client

Employee REST Controller

Update Employee

REST Client

PUT /api/employees

```
{
    "id" : 1,
    "firstName" : "Daniel",
    "lastName" : "Vega",
    "email" : "daniel.vega@luv2code.com"
}
```

Employee REST Controller

Update Employee

ID of employee to update
With updated info

REST Client

PUT /api/employees

```
{
    "id" : 1,
    "firstName" : "Daniel",
    "lastName" : "Vega",
    "email" : "daniel.vega@luv2code.com"
}
```

Employee REST Controller

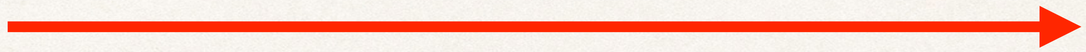
Update Employee

ID of employee to update
With updated info

REST Client

PUT /api/employees

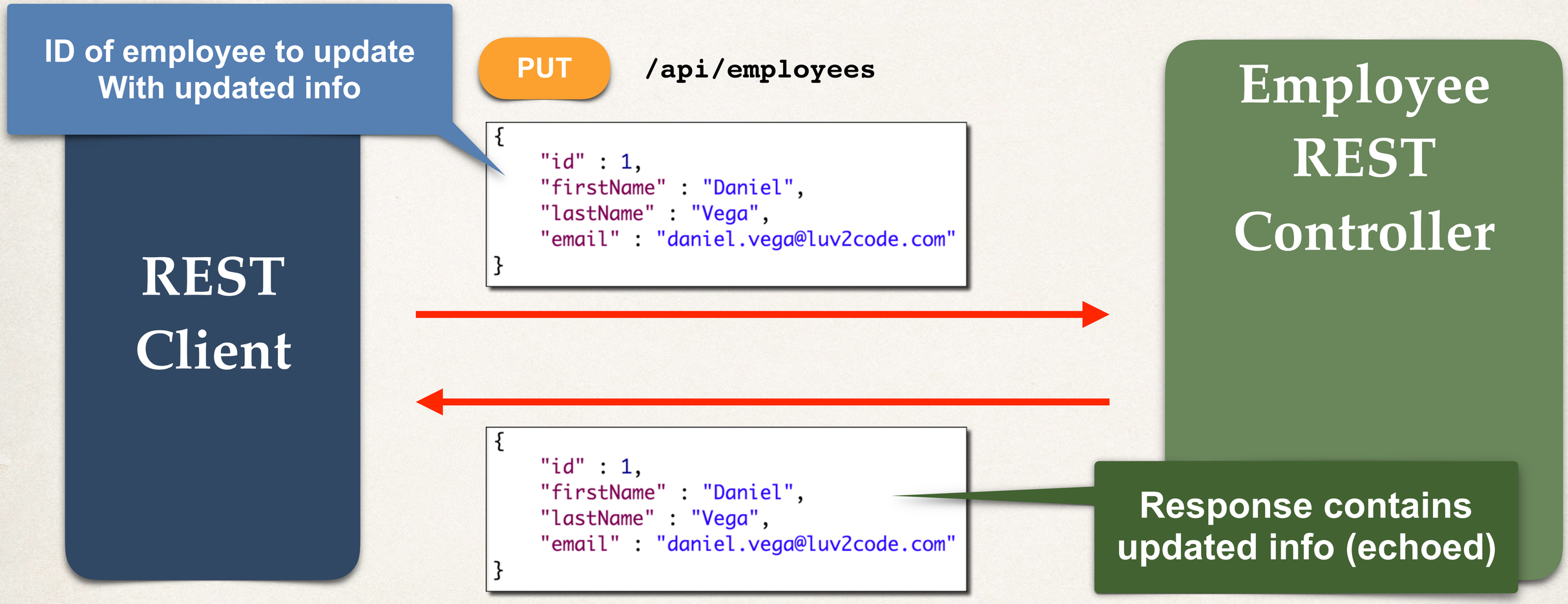
```
{
    "id" : 1,
    "firstName" : "Daniel",
    "lastName" : "Vega",
    "email" : "daniel.vega@luv2code.com"
}
```



```
{
    "id" : 1,
    "firstName" : "Daniel",
    "lastName" : "Vega",
    "email" : "daniel.vega@luv2code.com"
}
```

Employee REST Controller

Update Employee

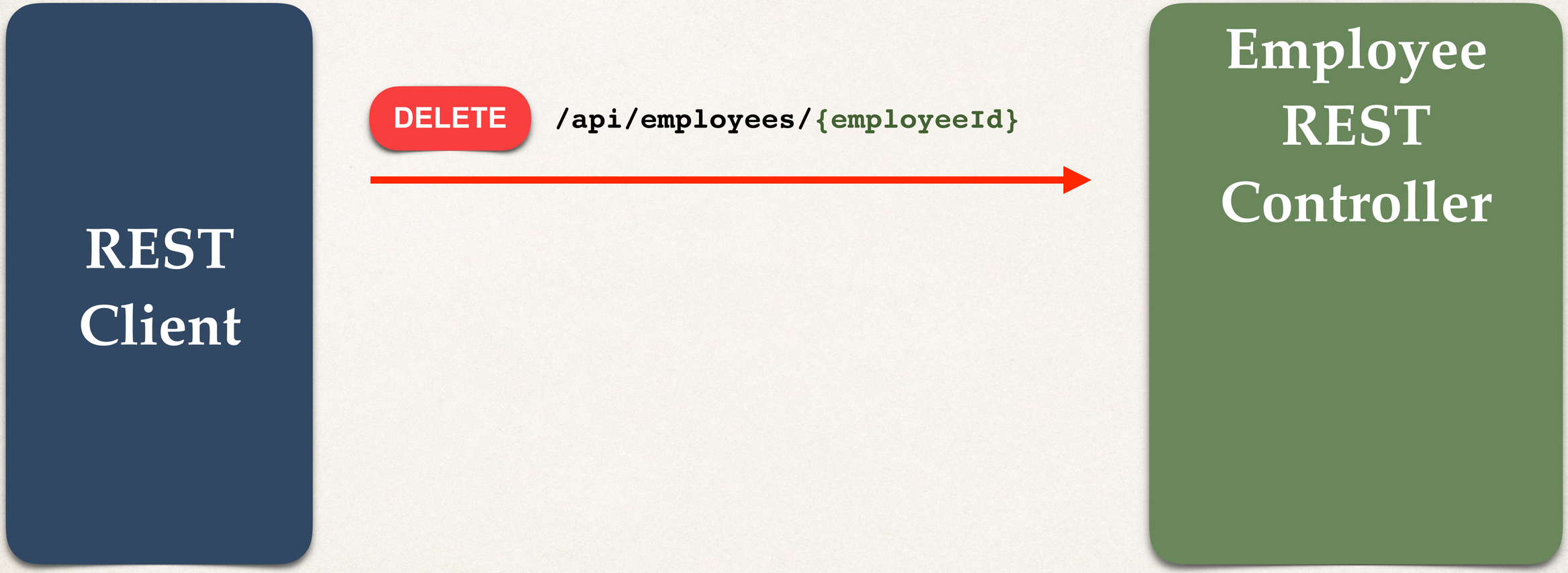


Delete Employee

REST Client

Employee REST Controller

Delete Employee



Delete Employee