## Introduction

The projects are designed to have progressive development, starting with a simple backend to let the student be focused on user interface design (UI), these interfaces can be as complex as the student wants to develop using skills learned in the bootcamp.

Once basic requirements have been completed and user interface has been developed for applications, the student has the option to create new endpoints to add new features to the application.

## Development

#### Stage 1 (base development)

**Backend:**

1. Read endpoint: To get data from database it will be returned on JSON format to the client.

**Frontend**:

1. UI is open design development. Design and implementation depends on the students.
2. Create API calls needed to get all data from the server.

#### Stage 2 (New features)

**Backend**:

Possible endpoints to develop:

1. Endpoint to insert new entries to the DB.
2. Endpoint to update items on the DB.
3. Endpoint to delete an entry on the DB.

**Frontend:**

1. Get every together, use the new endpoints developed and add the new features to the IU.

#### Stage 3 (Optional)

1. Add a search function on your application.
2. Add sort functionality to the lists on your application.

## Project

### Dog Rescue

If someone finds a lost puppy on the street and it has DogRescue’s tag, its tag’s id can help to find the owner. DogRescue web application enables people who find a dog with a tag to send an alert to the owner using the search engine on DogRescue website.

**Database:**

Table: puppies

Fields: id, name, breed, age, image\_url, created\_at

Table: alerts

Fields: id, puppy\_id, created\_at, deleted\_at

**Stage 1:**

*Backend*

Endpoint: GET /search/:puppy\_id

Description: Endpoint to get a puppy profile.

Suggested response

*HTTP 200*

{

id: int,

name: string,

breed: string,

age: int,

image\_url: string

}

*FrontEnd*

UI: Develop your own Dog Rescue website, be creative using all tools that you learned at the bootcamp.

Data source: Use endpoints developed by you to get data needed.

**Stage 2:**

Feature: Alert.

Description: Enable your application to add or remove an alert on a puppy profile.

*Backend*

Endpoint: POST /alert/:puppy\_id

Description: Add a alert on a puppy profile.

Suggested response: *HTTP 201*

Endpoint: DELETE /alert/:puppy\_id

Description: Remove an alert from a puppy profile, consider soft delete.

Suggested response: *HTTP 204*

*FrontEnd*

UI: Add the new functionality to you UI, and keep alert signal on the dog profile.