

Programming Boot Camp Learning Phase #4

Bubble Basic #2

2021/11/20

事前準備

- 本日は前回作成したペットの健康管理アプリに、デザインやロジックを追加していきます
- 本日の講義用に多少手を加えていますので、開始時点をそろえるため、こちら側で用意したアプリケーションを複製したものを利用してもらいます
- 複製したアプリケーションを配布しますので、Naotake KYOGOKU 宛てに、Bubbleのアカウントを作成したメールアドレスを伝えてください。



今日やること

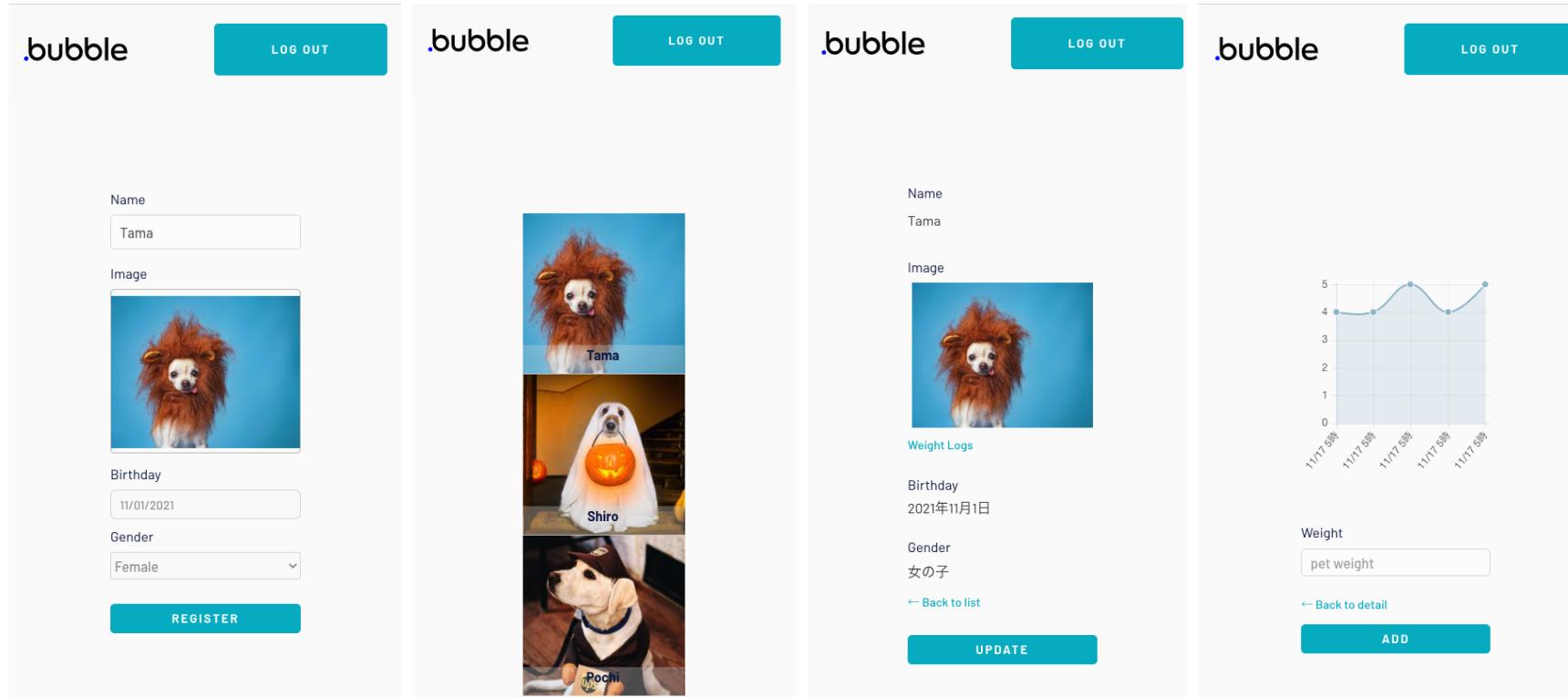
- 前回のおさらい
- デザインを作りこむ
- ロジックを作りこむ

前回のおさらい

- Bubble はビジュアルプログラミングツールとうたっており、画面からポチポチと操作して、見た目や動きをプログラミングできるツールです。
- Webアプリケーション前提であり、ディスプレイサイズにあわせた対応をいれスマホやPCに対応させます。
- 前回お休みの方はこちらの資料でキャッチアップしましょう
 - <https://github.com/GuildWorks/titech-2021/tree/master/docs/Bubble1>

前回のおさらい

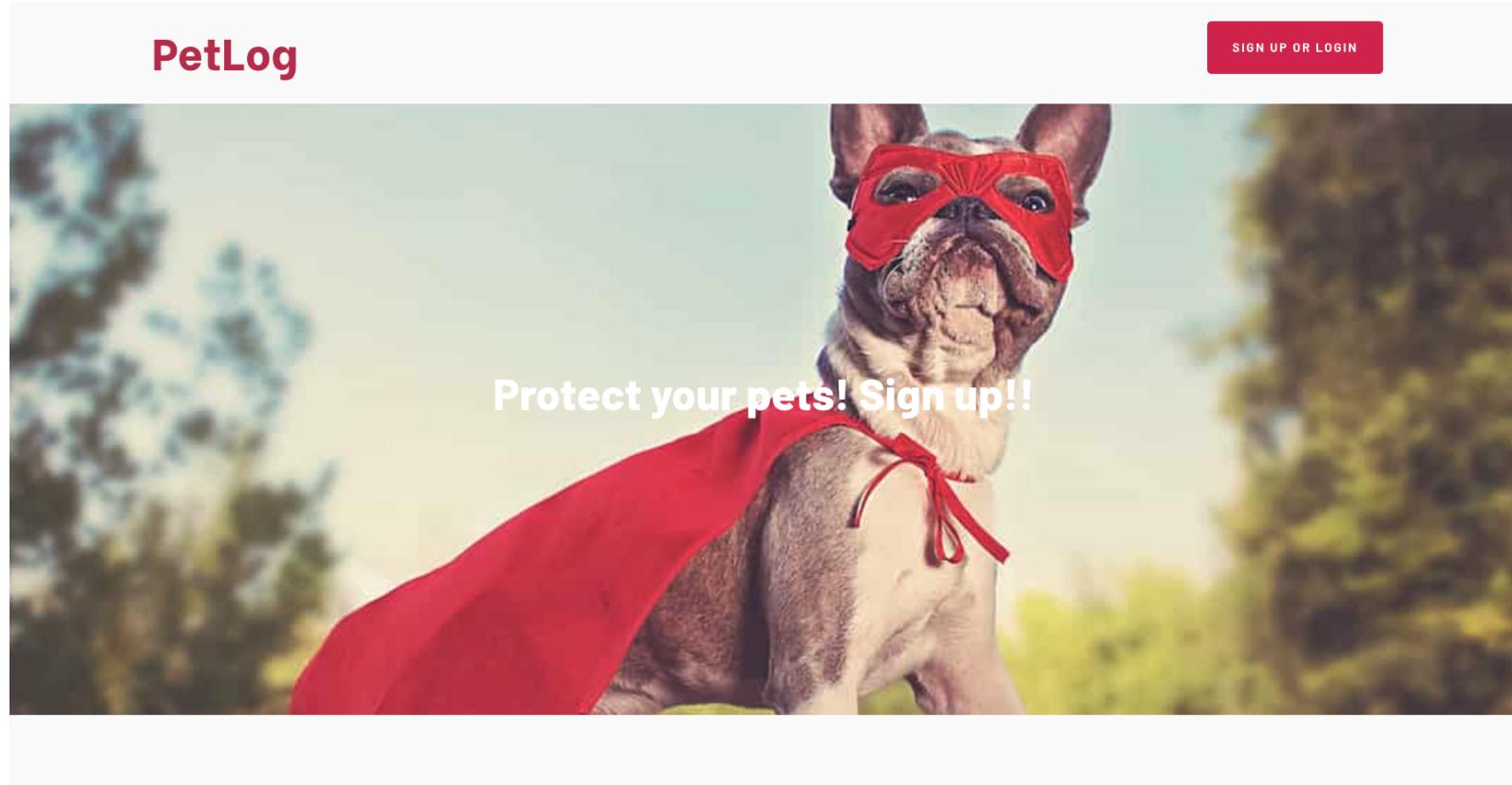
ペット管理アプリケーションのペットの登録、一覧、詳細、体重記録の画面を作りながら、Bubbleの基本である、Design/Workflow/Dataの使い方を学びました。



今回はここに対して、デザインやロジックを作り込んでいきます

最終的にはこんな感じになります

トップページ



ペットリスト

PetLog

LOG OUT



PET REGISTER

PetLog

LOG OUT



ペット詳細

PetLog

[← Back to list](#)

Image

Name (Initial)
pochi (P)

Birthday
2019年11月19日

Age (As Dog/Cat)
2 (24)

Gender
女の子

Latest Weight
3kg

UPDATE

Weight Logs

Date	Weight (kg)
11/17 7時	4.0
11/17 7時	4.0
11/17 7時	4.8
11/17 7時	3.0
11/17 7時	4.8
11/17 7時	3.0

PetLog

[← Back to list](#)

Image

Name (Initial)
pochi (P)

Birthday
2019年11月19日

Age (As Dog/Cat)
2 (24)

Gender
女の子

Latest Weight
3kg

UPDATE

Weight Logs

アドバイザー用ペットリスト

PetLog					LOG OUT
Name	Gender	Birthday	Owner		
 kuro	Female	2021年11月4日	kim@guildworks.jp		
 tama	Male	2021年11月2日	kim@guildworks.jp		
 pochi	Female	2019年11月19日	kim@guildworks.jp		
 コロネ	Male	2019年10月1日	kyogoku+bubble_test@guildwo rks.jp		
 マロン	Male	2021年7月4日	kyogoku+bubble_test@guildwo rks.jp		

アドバイザー用ペット詳細

PetLog

[← Back to list](#)

Image



Name (Initial)
pochi (P)

Birthday
2019年11月19日

Age (As Dog/Cat)
2(24)

Gender
女の子

Latest Weight
3kg



Date	Weight (kg)
11/17 7時	4.0
11/17 7時	4.0
11/17 7時	4.0
11/17 7時	5.0
11/17 7時	3.0
11/17 7時	5.0
11/17 7時	3.0

では、さっそくはじめていきましょう。

まずはデザインを作りこんでいきます

デザインの作り込みでやること

- ディスプレイサイズに合わせた画面をつくろう
 - レスポンシブウェブデザインという手法を使って、ディスプレイサイズに合わせて以下のように見た目を制御します
 - 伸びる／縮む
 - 折り返す／折り返さない
 - 表示する／表示しない
- Styleを使ってみよう
 - Styleを編集・追加したり、個別にスタイルをあてます

ディスプレイサイズに合わせた画面をつくろう

- WebアプリケーションはPCやタブレット、スマートフォンといった様々な端末で利用されます。
- 端末毎にディスプレイサイズが違うのですが、それらに対応する手法としてレスポンシブウェブデザインというデザイン手法があります
- 画面サイズに応じて、要素が伸びる／縮む、折り返す／折り返さない、表示する／表示しないといった見た目の切り替えを行うという手法になります。
- Bubbleは標準でレスポンシブウェブデザインの機能をそなえており、意識しなくとも対応かれています。

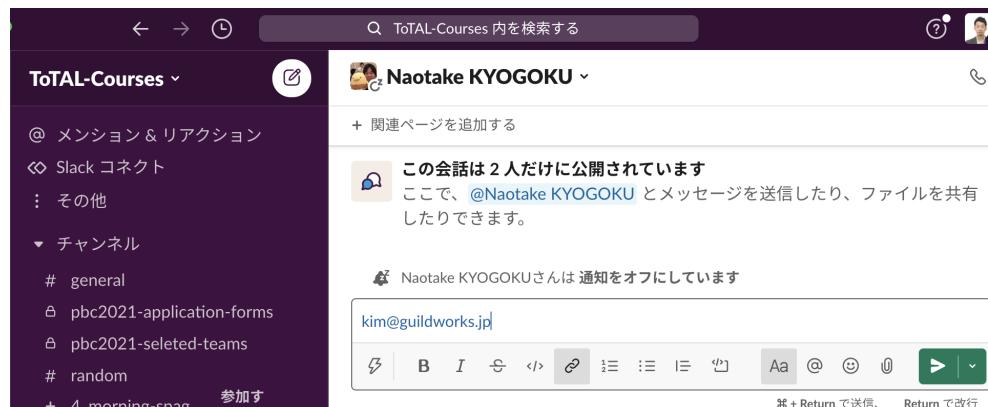
レスポンシブウェブデザインの適用の仕方

- レスponsivewebdesignを考えるときには、まずはPC画面を作成し、ディスプレイサイズが小さくなつていったときの表示ルールを付け足すことが多いです。
- スマートフォンでの利用場面が主となるような場合には、スマートフォンの画面を先に作成して、PC表示をあとから整えるということもあります。

さっそく体験してみましょう。

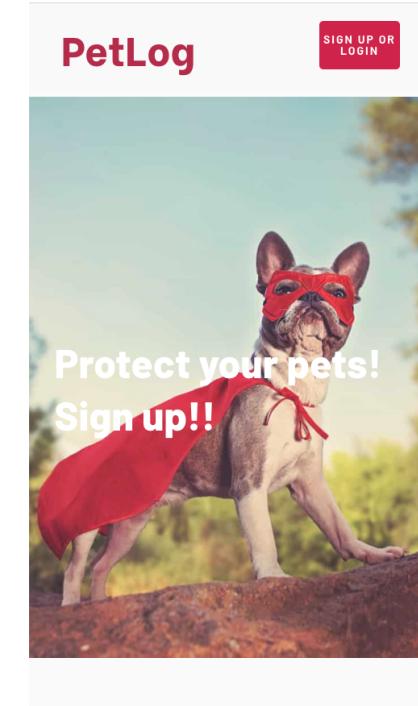
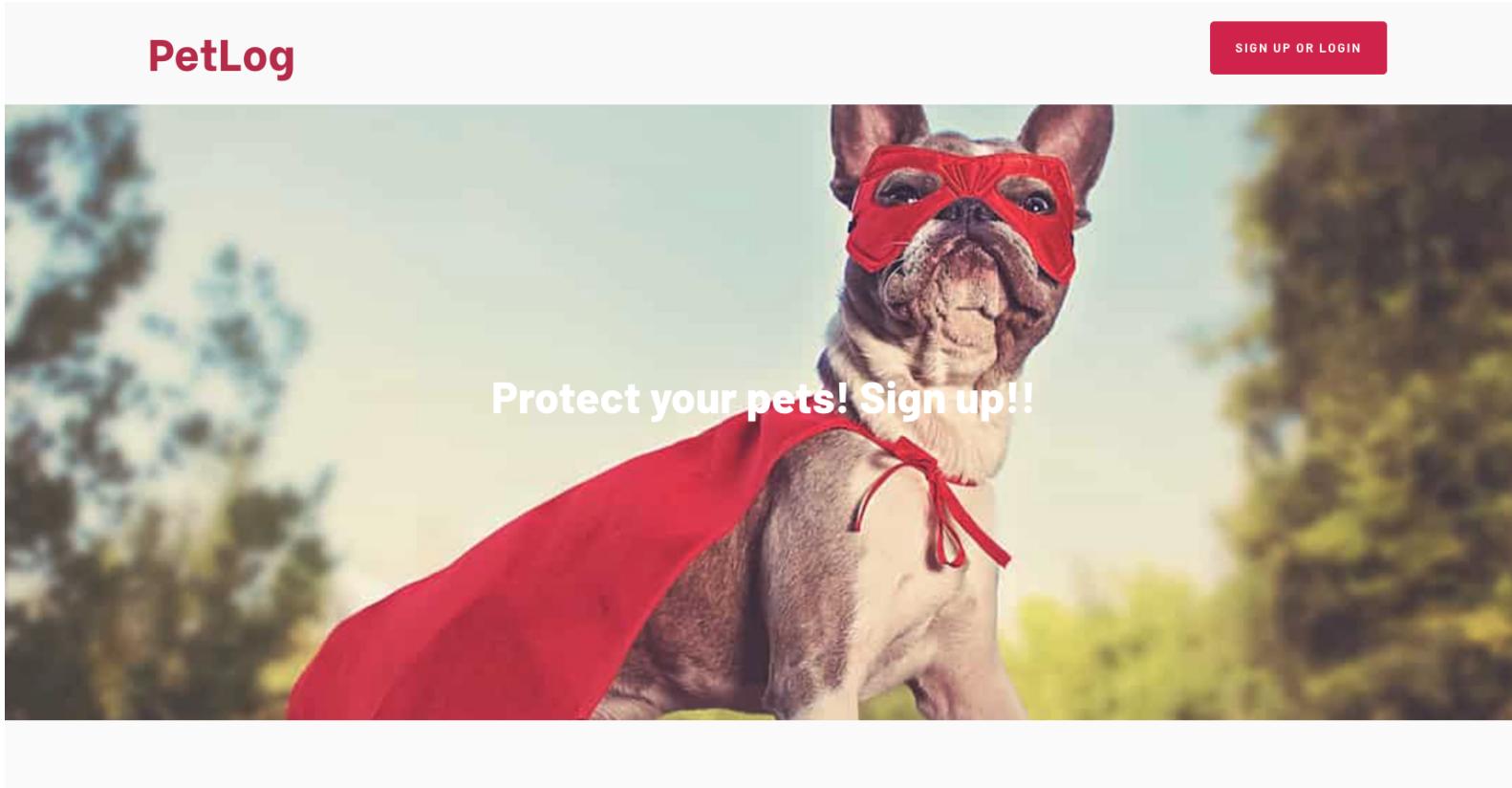
(再確認) 事前準備

- 本日は前回作成したペットの健康管理アプリに、デザインやロジックを追加していきます
- 本日の講義用に多少手を加えていますので、開始時点をそろえるため、こちら側で用意したアプリケーションを複製したものを利用してもらいます。
- 複製したアプリケーションを配布しますので、Naotake KYOGOKU 宛てに、Bubbleのアカウントを作成したメールアドレスを伝えてください。



トップ画面に画像とキャッチを配置します

トップ画面に画像とキャッチを配置して、レスポンシブ対応をいれていきましょう。



画像準備

私はこちらの画像を使います。

<https://raw.githubusercontent.com/GuildWorks/titech-2021/master/docs/Bubble2/materials/hero.jpg>

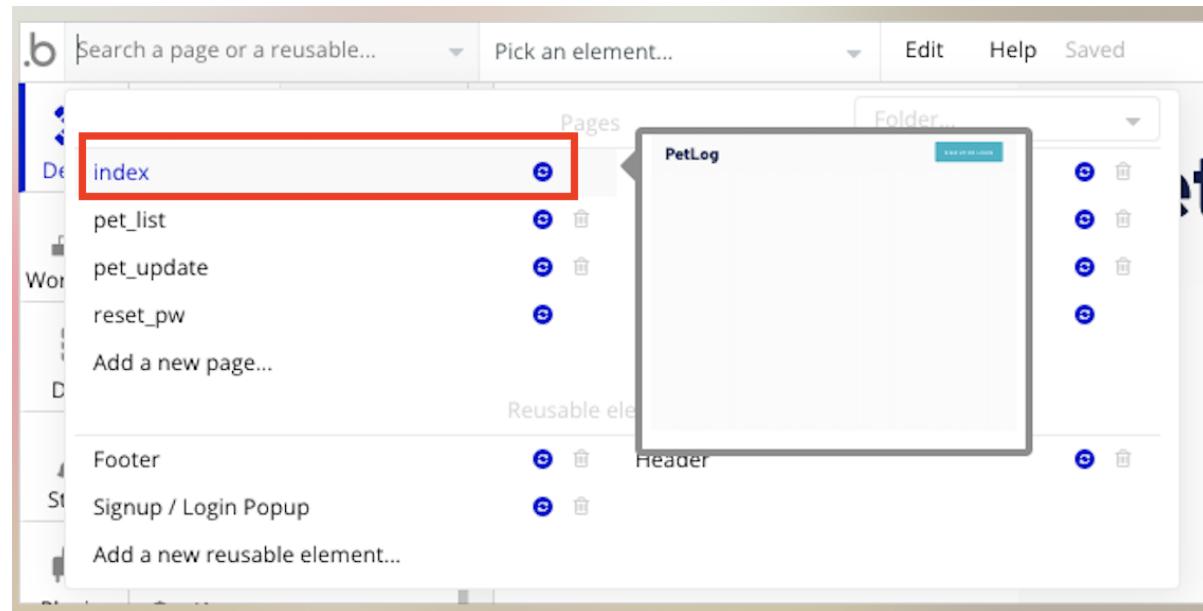
好きな画像を利用してもらっていいですが、大きく表示するので解像度の高いものを選んでください。



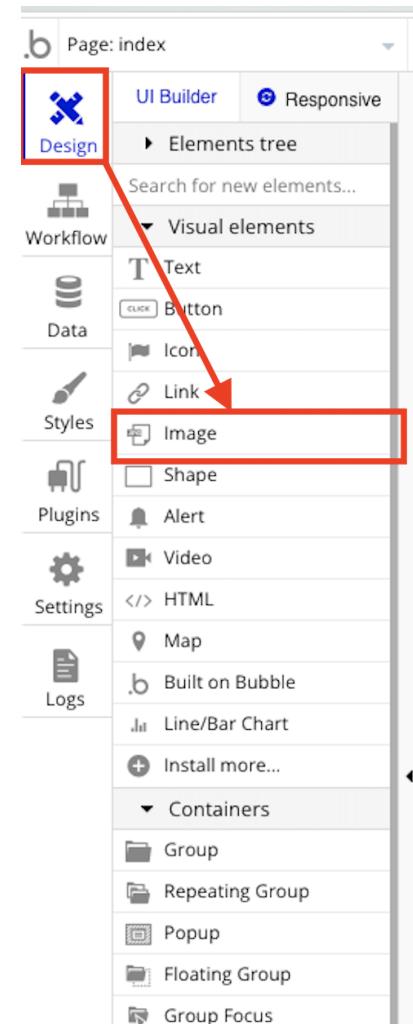
トップページに画像を配置する

トップページに移動してください。手順は覚えていませんか。

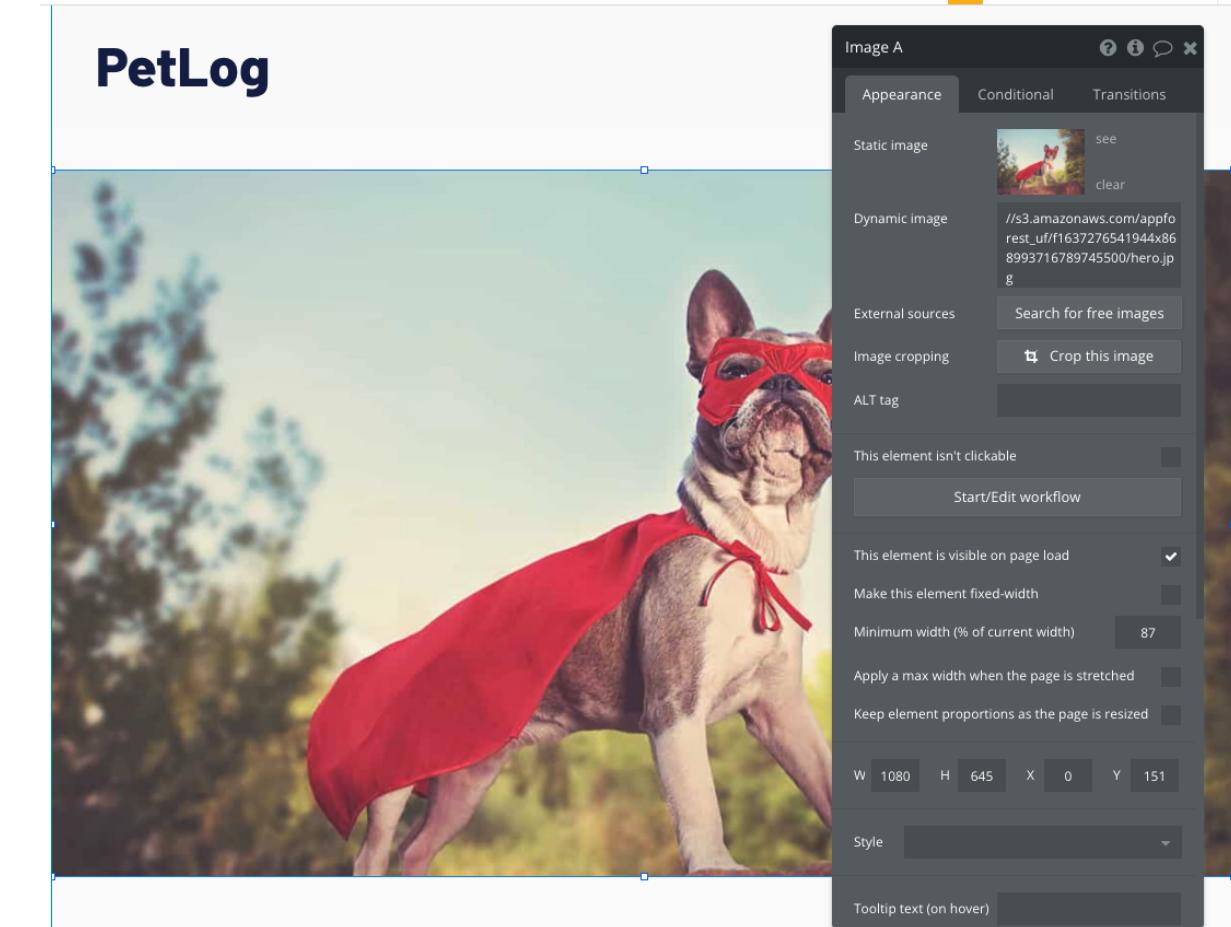
- 画面左上のbロゴの右をクリックして、indexページを選ぶ。



- 左メニューから、
Design > Image の順にクリック
する。
- 画面の中の真ん中あたりに適当な
大きさで配置する。
 - 配置したい場所で配置したい
大きさになるように、ドラッ
グする。
- 画像Elementが置かれたらOK（ま
だ画像は空の状態）。

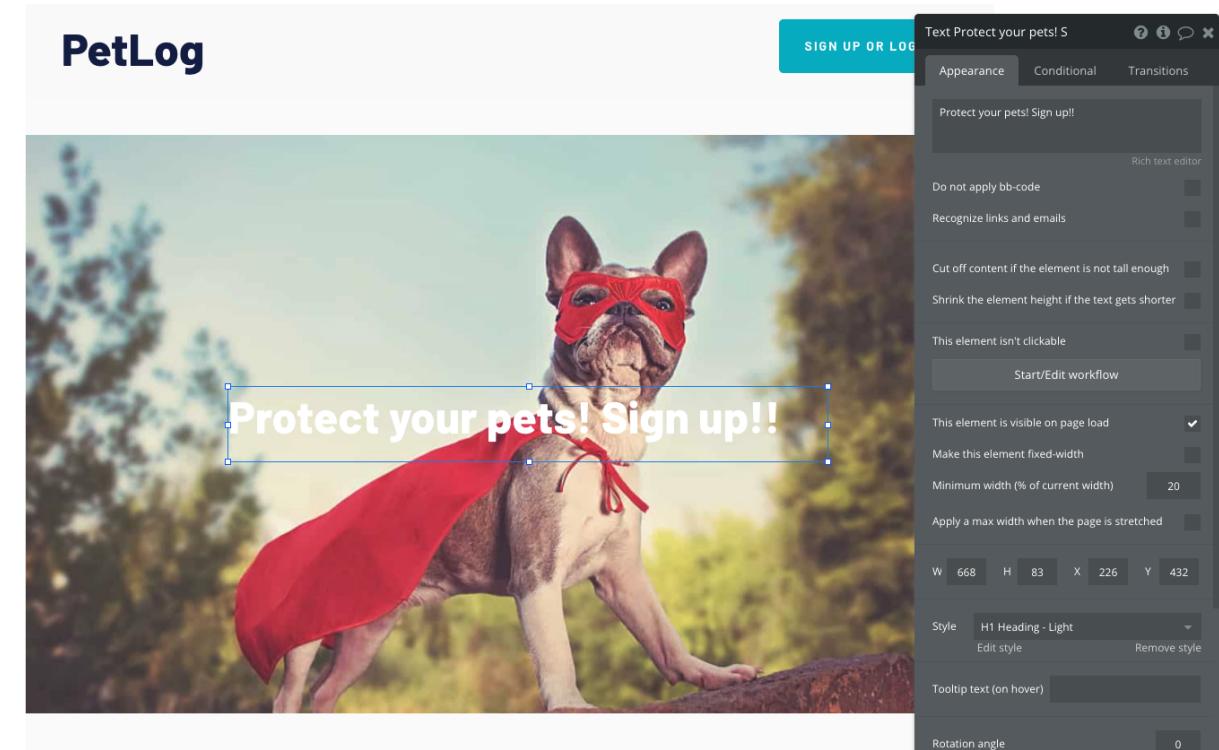


- 画像Elementの右側に設定用のポップアップがでているので、設定をいれていく。
- **Static Image** に用意した画像をアップロード。
- 画像を左右いっぱい今まで引き伸ばし、上下も好きな具合に整える。
 - 犬ちゃんマンの場合は
W:1080、H:645くらいがよさそう



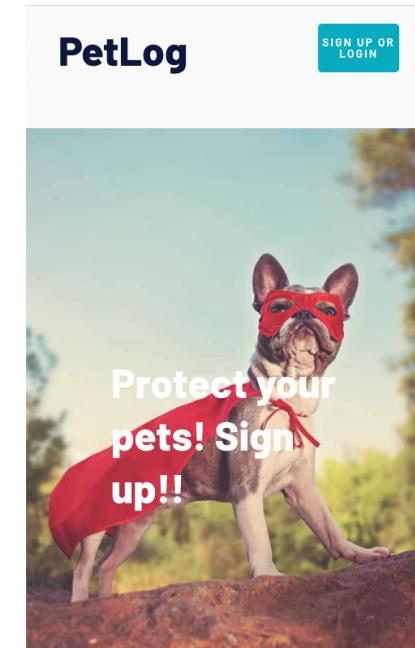
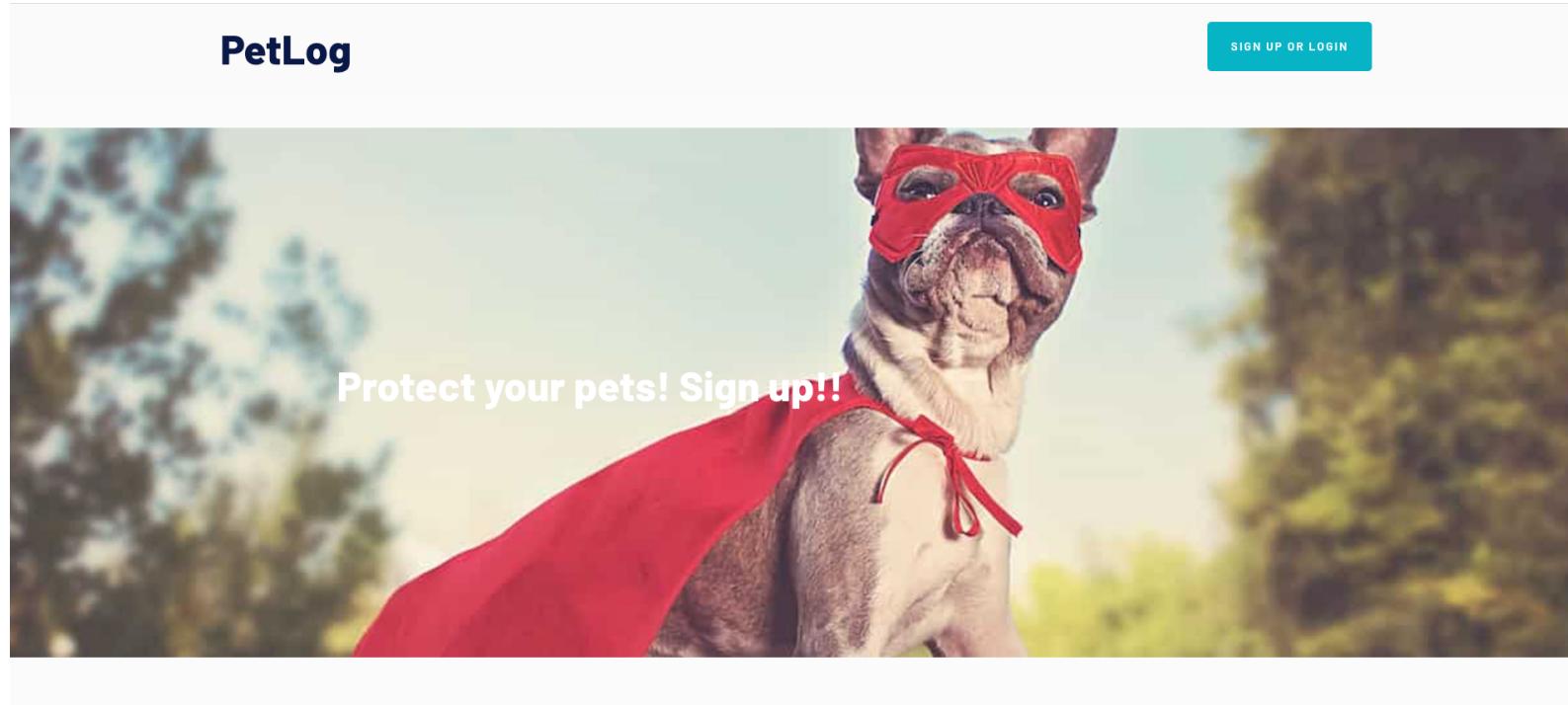
次にテキストを配置します。

- 左メニューから **Text** をクリックして、画面に中央に配置。
- テキストElementの右側に設定ポップアップがでているので、設定をいれていく
 - 最上部のテキストエリアに好きなキャッチコピーを入れる。
 - 私は、 **Protect your pets! Sign up!!** といれます。
 - Styleに **H1 Heading - Light** を指定する



ではプレビューしてみましょう。

- 画面右上の `Preview` リンク か、ショットカット (Windows : `Ctrl+p` 、 Mac : `Cmd+p`)



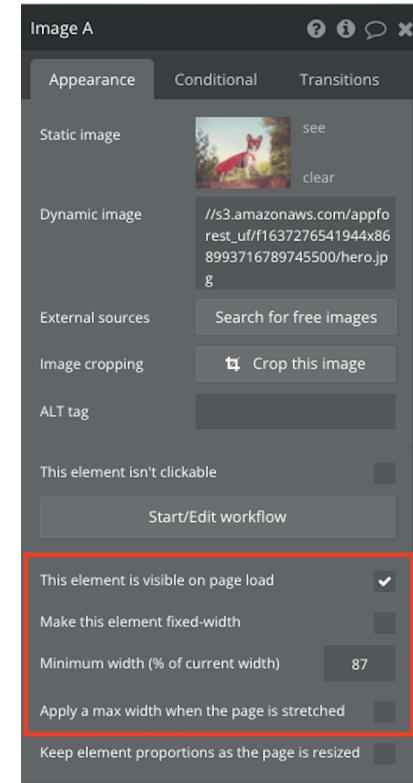
特にレスポンシブ対応をいれていませんが、勝手にいい感じにしてくれていますね。Bubbleがレスポンシブに標準で対応してくれているためです。

少し解説します。

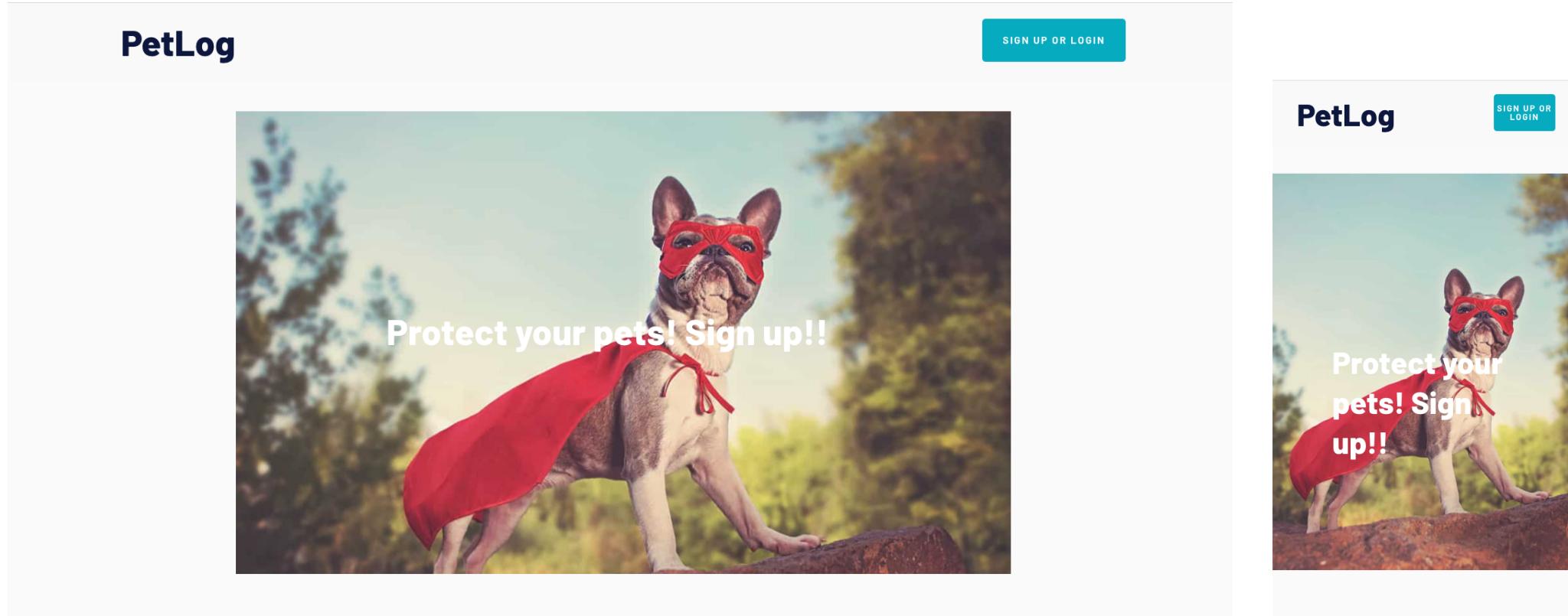
画像やテキストの設定を見てみましょ
う。中段あたりにこのような設定があり
ます。

- Make this element fixed width
- Minimum width
- Apply a max width when the page is stretched

要素毎にこれらの設定を使いながら意図
した見た目になるように調整していきま
す。試しに設定を変えて表示してみま
す。

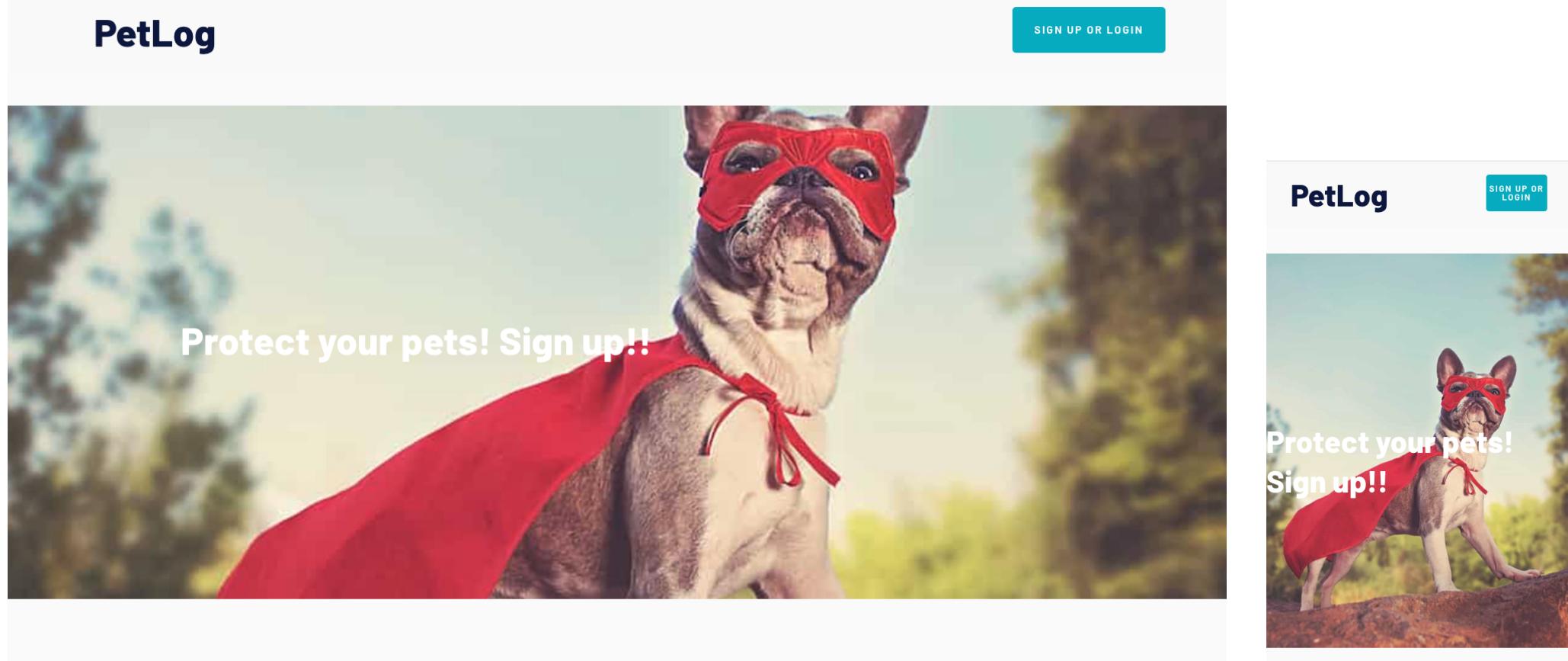


`Make this element fixed width` : これはこのelementの横幅を固定するかどうかという設定です。チェックが入っていない場合は、画面幅にあわせて伸び縮みします。試しに画像の設定で、チェックをつけてプレビューしてみましょう。



画面幅を大きくしても画像elementの幅は変わらなくなりました。（チェックははずしておいてください。）

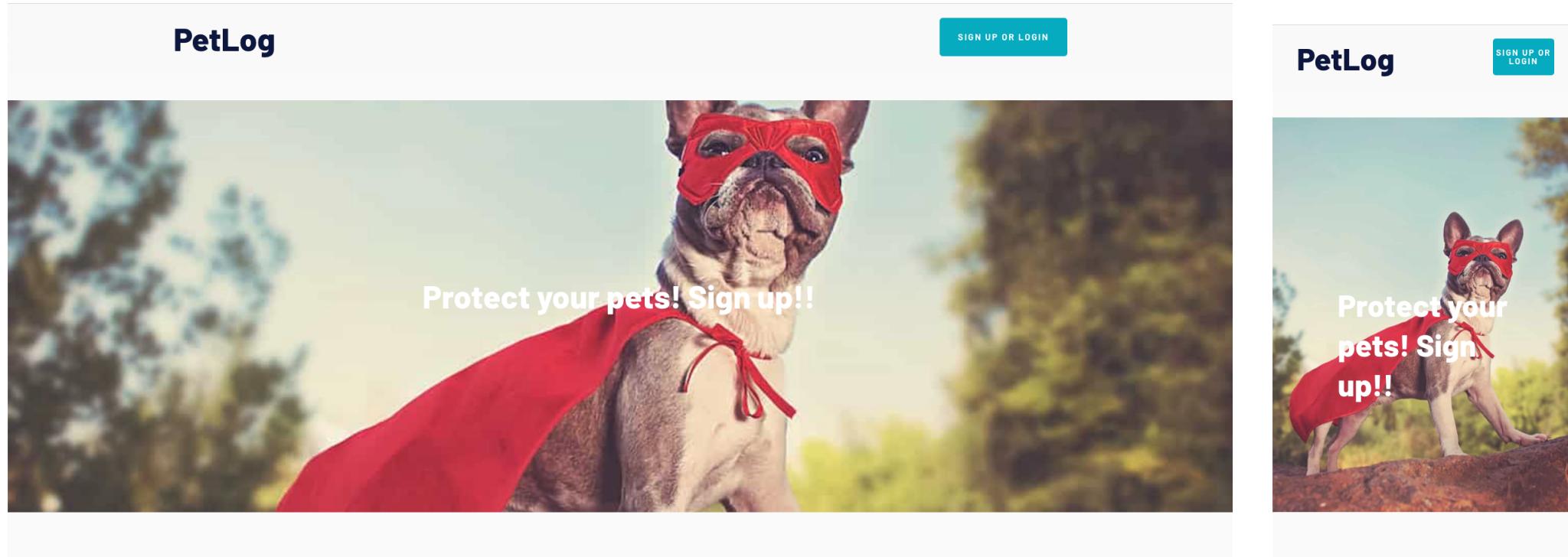
Minimum width : これはelementの最小幅です。画面幅を狭めていくと、合わせてelementも小さくなっていくのですが、ここで指定したサイズ以上には小さくなりません。試しにテキストの設定で80%に指定してみてプレビューしてみてください。



画面幅を小さくし他時に、変更前ほどテキスト表示に使っている幅が狭まらなくなりました。 (20%に戻しておいてください。)

Apply a max width when the page is stretched : これはelementの最大幅です。画面幅を広げていくと、合わせてelementも伸びていきますが、ここで指定したサイズ以上に大きくなりません。

試しにテキストの設定で、チェックをつけて、100%にしてみください。



画面を大きくしていくと、elementが伸びないので、文字が左によらなくなりました。
この設定はそのままで構いません。

次に、画面幅を変えた場合の配置についても解説しておきます。

人によってキャッチコピーは、画面を大きくすると左や右にずれているのではないかと思います。

画面幅が伸びたときのelementの配置は、元の相対位置できます。

もともと真ん中に見えていて、気にならないくらいでも右や左にずれないと、画面が大きくなったときにそれが目立つようになります。

確実に真ん中に配置したい場合は、先週やった右クリック（もしくは二本指でタップ）して、`Center horizontally` を指定しましょう。

ここですね。

