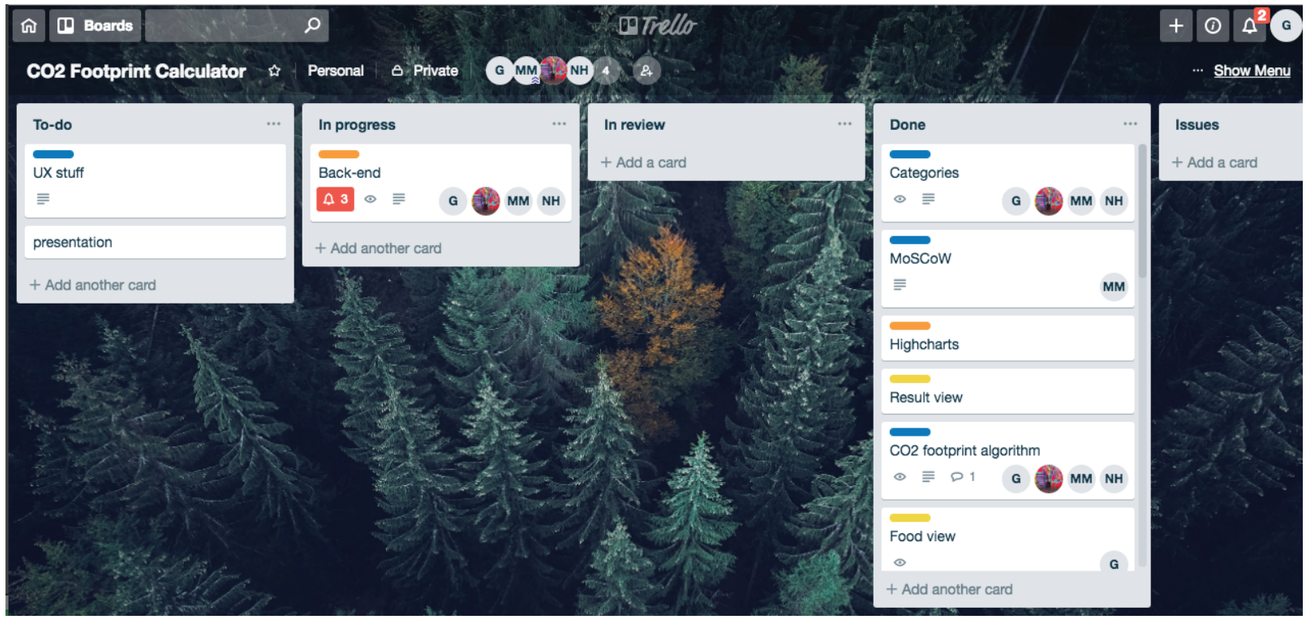
CO2 PRINT


We believe that there are people who want an online application that they can use to track their CO2 footprint and use this to encourage better practice.

Planning:

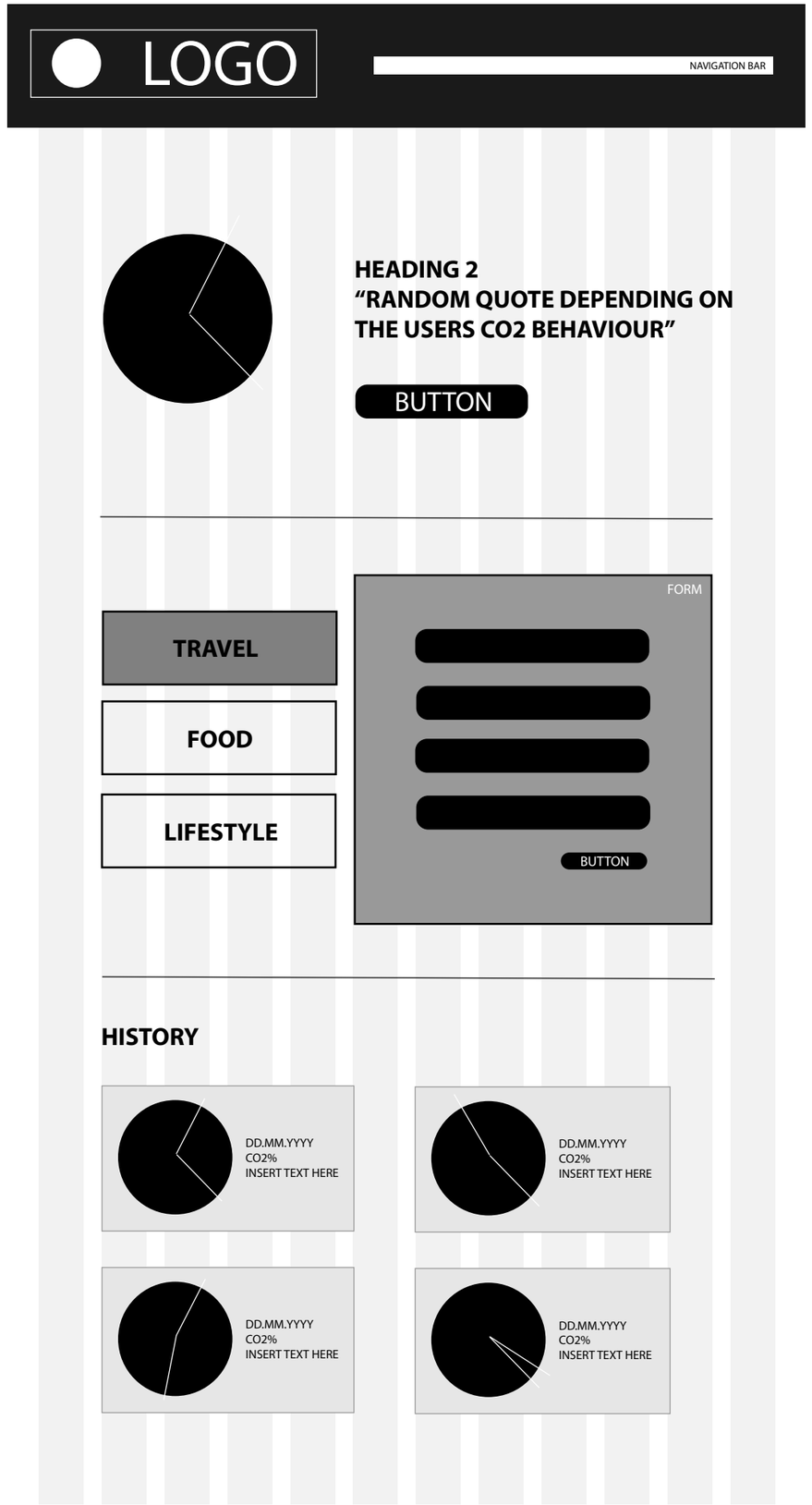


MoSCoW - CO2 Calculator

File Edit View Insert Format Data Tools Add-ons Help Last edit was made 6 days ago by Monica M

Must

	A	B	C	D	E
1	**Must**	**Should**	**Could**	**Won't**	
2	Log trips, stuff, food	Have personalized messages	Be able to offset points	Use an authentic calculation method	
3	Calculate CO2 footprint	Have a view that compares scores by category	Have pictures and photos	Get into detail with categories	
4	Display score		Send email notifications		
5	Have multiple form views				
6	Implement charts API				
7					
8					



Code:

```
PubSub.subscribe('PublishView:final-result', (event) => {
  this.all = event.detail;
  this.render();
  this.randomize(this.food, this.travel, this.lifestyle);
  this.chart();
  });
```

```
ResultView.prototype.randomize = function (food, travel, lifestyle) {

  const randomQuote = new Recommend;
  const maxNumber = Math.max(food, travel, lifestyle)
  if (maxNumber === food) {
    randomQuote.generate("food")
  } else if (maxNumber === travel) {
    randomQuote.generate("travel");
  } else if (maxNumber === lifestyle) {
    randomQuote.generate("lifestyle");
  } else {
    randomQuote.generate("lifestyle");
  }

  return randomQuote;
};
```

```
ResultView.prototype.chart = function () {

Highcharts.theme = {
  colors: ['#CD5C5C', '#20B2AA', '#3CB371', '#4682B4'],
  title: {
          style: {
              color: 'white',
              font: '22px "Raleway",sans-serif',
              textTransform: 'uppercase',
              marginTop: 70
          }
        },
  legend: {
        itemStyle: {
            color: 'white',
            font: "'Raleway', sans-serif",
            textTransform: 'uppercase'
        }
      }
};

Highcharts.setOptions(Highcharts.theme);

  const myChart = document.createElement("form")
  myChart.id = "result-view"
  const theChart = Highcharts.chart(myChart, {
    chart: {
        plotBackgroundColor: null,
        plotBorderWidth: null,
        plotShadow: false,
        type: 'pie',
        backgroundColor: '#BBCBCB'
```

Code:

```
CreateTravelForm.prototype.createSubmitButton = function () {
  const submitButton = document.createElement("input");
  submitButton.classList.add("submit");
  submitButton.type = "submit"
  submitButton.value = "SUBMIT"

  return submitButton;
};

  CreateTravelForm.prototype.createRadioButton = function (name, value, id) {
    const detail = document.createElement("input");
    detail.classList.add('checkmark')
    detail.type = "radio"
    detail.name = `${name}`
    detail.value = `${value}`
    detail.id = `${id}`

    return detail
  };

CreateTravelForm.prototype.createLabel = function (textContent, id) {
  const detail = document.createElement("label");
  detail.textContent = `${textContent}`;
  detail.id = `${id}`;

  return detail;
};

CreateTravelForm.prototype.handleSubmit = function (formInput) {
  formInput.addEventListener("submit", (event) => {
    event.preventDefault();
    const form = event.target;
    const answerArray = this.getValues();
    PubSub.publish("TravelModel:send-values-array", answerArray);
    form.reset();
  });
};

CreateTravelForm.prototype.getValues = function () {
  let carAnswer = document.querySelector('input[name="car-miles"]:checked').value;
  let busAnswer = document.querySelector('input[name="bus-miles"]:checked').value;
  let bikeAnswer = document.querySelector('input[name="bike-miles"]:checked').value;
  const answerArray = [carAnswer, busAnswer, bikeAnswer];
  return answerArray
};
```

```
CreateTravelForm.prototype.createForm = function () {
  event.preventDefault();
  this.container.innerHTML = ""

  const travelForm = document.createElement("form");

  travelForm.classList.add('travel-form');
  travelForm.id = ("travel-form");

  header = document.createElement("h2");

  header.textContent = "ENTER TRAVEL DETAILS:"
  travelForm.appendChild(header);

  const carQuestion = this.createCarQuestion();
  travelForm.appendChild(carQuestion);

  const publicTransportQuestion = this.createPublicTransportQuestion();
  travelForm.appendChild(publicTransportQuestion);

  const nonFootprintQuestion = this.createNonFootprintQuestion();
  travelForm.appendChild(nonFootprintQuestion);

  const submitButton = this.createSubmitButton();
  travelForm.appendChild(submitButton);

  this.handleSubmit(travelForm);

  const newForm = this.container.appendChild(travelForm)
  return newForm;
};
```

What we learnt:

Any Questions?