

A large red speech bubble graphic with a white outline, pointing downwards. It contains the text 'Andrew Zoric' and 'UI/UX Portfolio'. The background features faint, overlapping circular lines in light gray and white, some solid and some dashed.

**Andrew Zoric**

**UI/UX Portfolio**

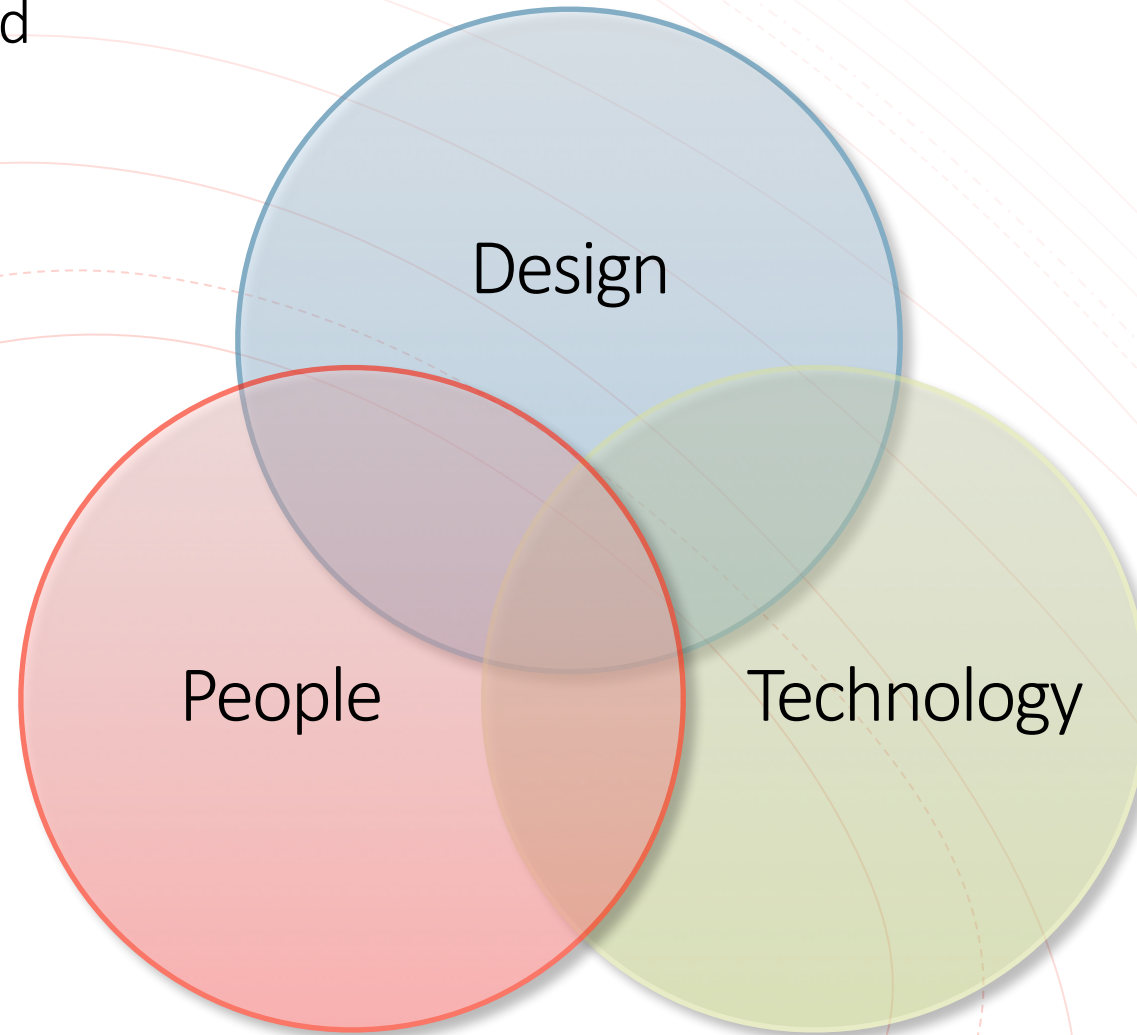
The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large red shape, resembling a speech bubble or a callout box, is centered on the page. It has a rectangular top and a pointed bottom. The text is white and centered within this red shape.

# Part I

UX Philosophy

The user interface is based on 3 elements:

- Design
- Technology
- People

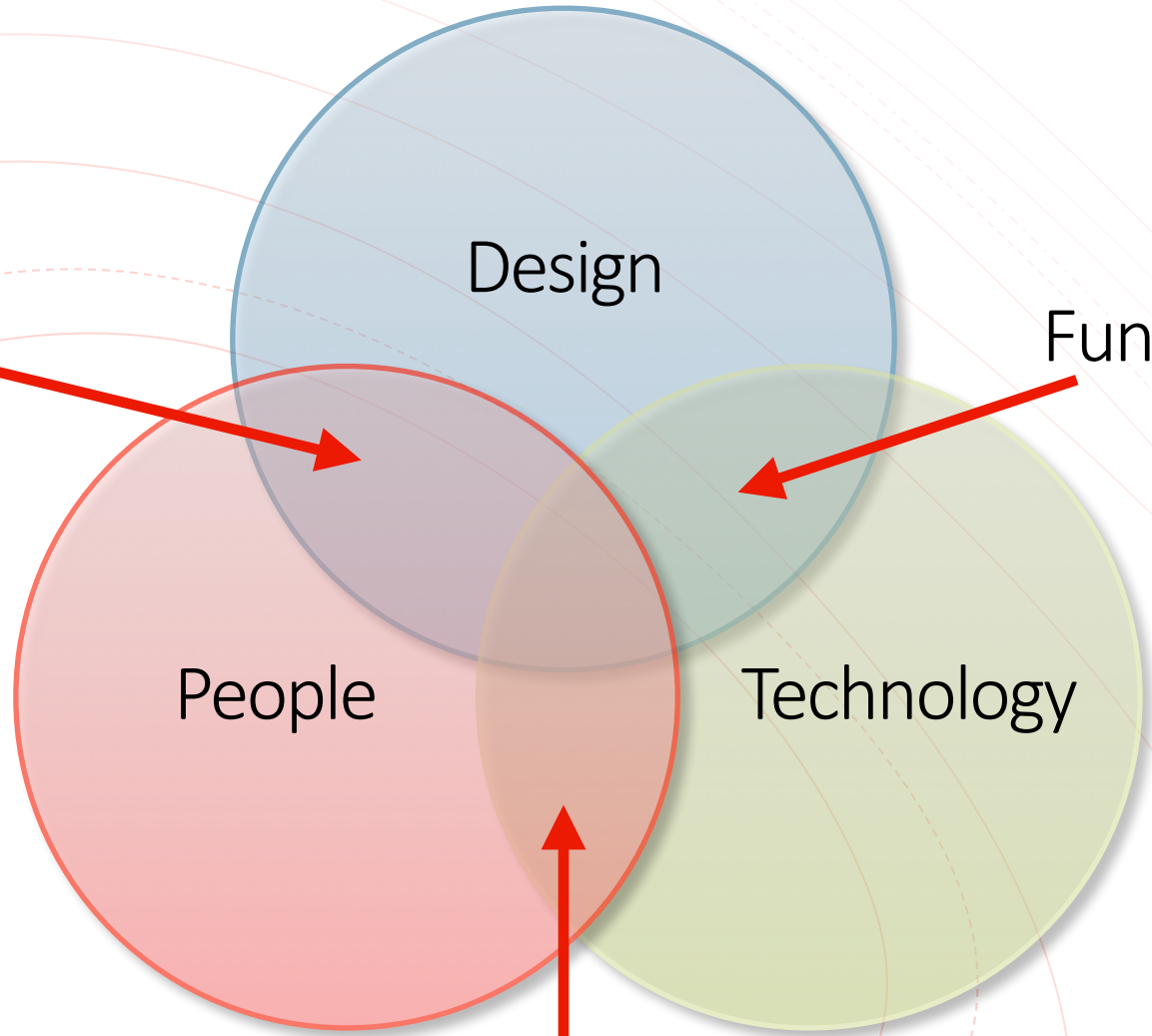


Resulting in combinations

- Form
- Function
- Interaction

Form

Functionality



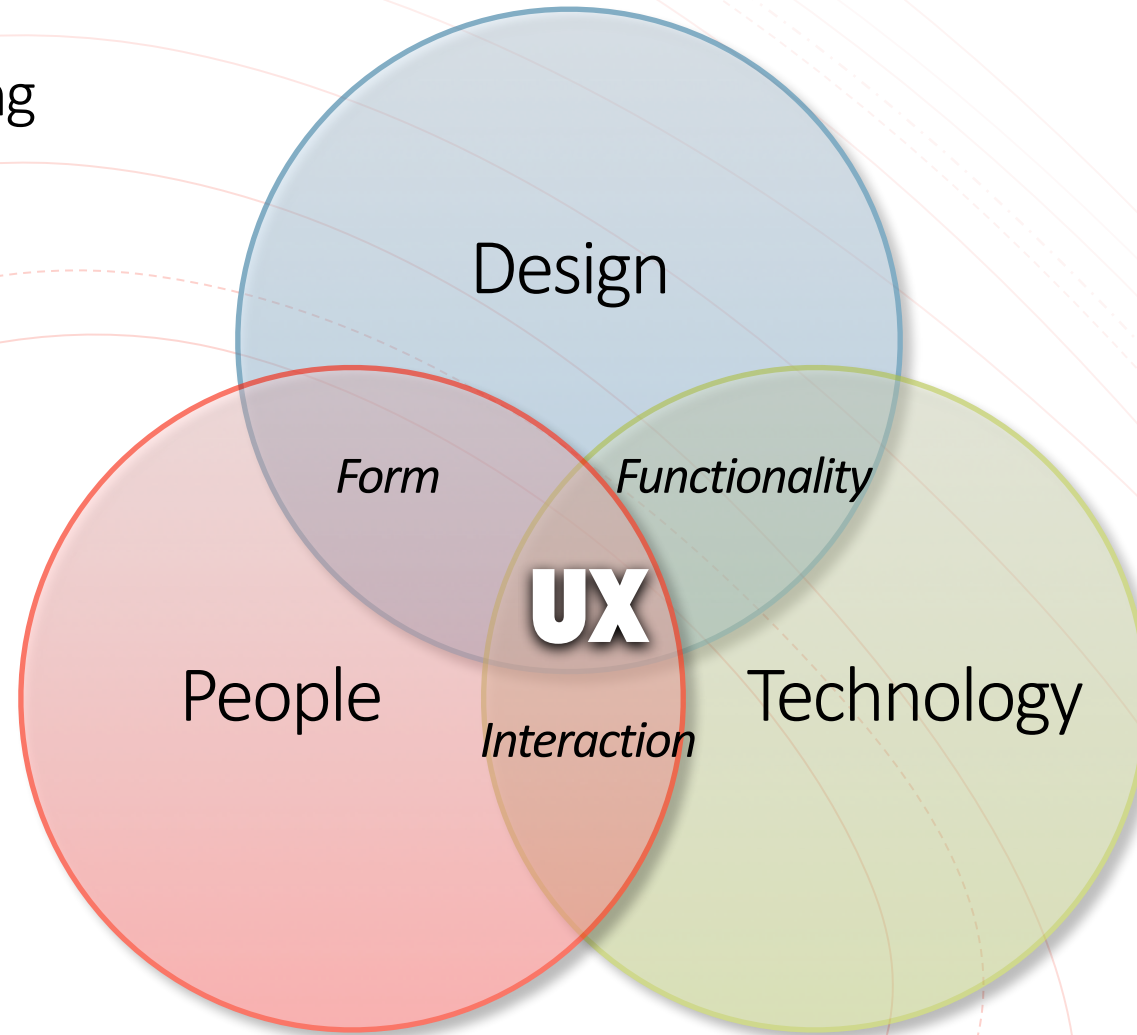
People

Technology

Design

Interaction

UX is the culmination of  
all these elements working  
together



# Part 2

## Examples: OnePlace

An executive portal for managing widespread IT assets.  
Designed and developed by me.

Tech Stack: React, Redux, SASS, C# (back end)

The screenshot displays the Cargill OnePlace interface. The top navigation bar includes the Cargill logo, 'OnePlace', a search icon, and a user profile icon labeled 'HS'. A left sidebar contains navigation options: Home, Graph (selected), Search, Health, Dashboards, Tags, Measures, Data Sets, Frameworks, and MAC Registration. The main content area is titled 'Graph : Columbus' and features a 'Currently Selected' panel on the left and an 'Asset Map' panel on the right. The 'Currently Selected' panel shows 'Name: Location' and 'Type: Location'. Below it, a 'Connections' section lists 'Location - Located In - (2)' with a green checkmark. The 'Asset Map' panel has tabs for 'Asset Map', 'Table View', 'Dashboard', and 'History'. It displays a branching tree starting from 'Columbus'. The tree branches into 'Location', 'Machine', 'Network', 'SourceOfContent', 'Stain', 'Internal Asset', and 'HPF31ABF'. The 'Location' node is expanded, showing two child nodes: 'HSW100 [SysID:1934]' and 'HSW100 [SysID:1935]'. Each node includes a pin icon and a plus icon for toggling its state.

## Branching Tree Interface

- Individual nodes clickable to toggle an open state.
- Open state reveals children, unless there are none in which a terminal node indicates there are no child nodes.
- Metadata for each node sets an icon for the type of node.
- A pin drop icon is clickable and recenters the entire graph with that node as the root.
- Originally planned as a vector-based D3.js object I was able to model the required behavior in JavaScript/CSS and HTML to achieve the desired effect, giving more control through the browser and integration with encapsulating React code.

**Cargill OnePlace** HS

**Graph : Columbus**

**Currently Selected**  
Name : SourceOfContext  
Type : RefersTo

**Connections:**  
SourceOfContext - Refers To - (3)

**Asset Map** | Table View | Dashboard | History

Columbus

- Located In: Location
- Located In: Machine
- Located In: Network
- Refers To: SourceOfContext (500 Internal Server Error)
- Located In: State
- Refers To: flexnet\_asset
- Located In: HPF31ABF

Support: -oneplace-team-all  
Version 0.1.0

## Branching Tree Interface

- Additional features shown.



Cargill OnePlace

### Graph : Hopkins

**Currently Selected**  
Name: HOPKINS  
Type: City

Version  
1

Createdon  
2020-10-26T19:06:27.276667

Createdby  
fct\_location

IsActive  
true

Batchid  
6607050-a006-4714-a047-30edf90eac4e

Updatedon  
2020-10-26T19:06:27.276667

Updatedby  
fct\_location

Sources  
[{"states":{"fct\_location":{"reportedOn":"2020-10-26T19:06:27.276667","isActive":true,"batchid":"6607050-a006-4714-a047-30edf90eac4e"},"Texnet\_asset":{"isActive":true,"batchid":"14-4e2ef1990-4200-833a-d01c1ed1939"},"scm\_asset":{"isActive":true,"batchid":"c88d9bf-805c-4fe2-8391-e9e0f046c156"},"Texnet\_fw":{"isActive":true,"batchid":"880e020-c79c-4479-51e7-41871f4d997"},"Texnet\_storage":{"isActive":true,"batchid":"e8b9d73-7d37-4076-b921-fc0c8e59145"}}]

Label  
city

Frequent Relations  
Loading...

Connections  
Loading...

Asset Map Table View Dashboard History

HOPKINS Loading...

## Branching Tree Interface

- Loading screen.
- Nodes animate out of their parent.
- Cog icon rotates while data is fetched.

Cargill OnePlace

HS

Graph : Columbus

Currently Selected

Name: State

Type: LocatedIn

Connections:

State - Located In - (2)

Asset Map

Table View Dashboard History

Columbus

Located In Location

Located In Machine

Located In Network

Refers To SourceOfContext

Located In State

locatedIn Nebraska

locatedIn Ohio

Refers To Internal Asset

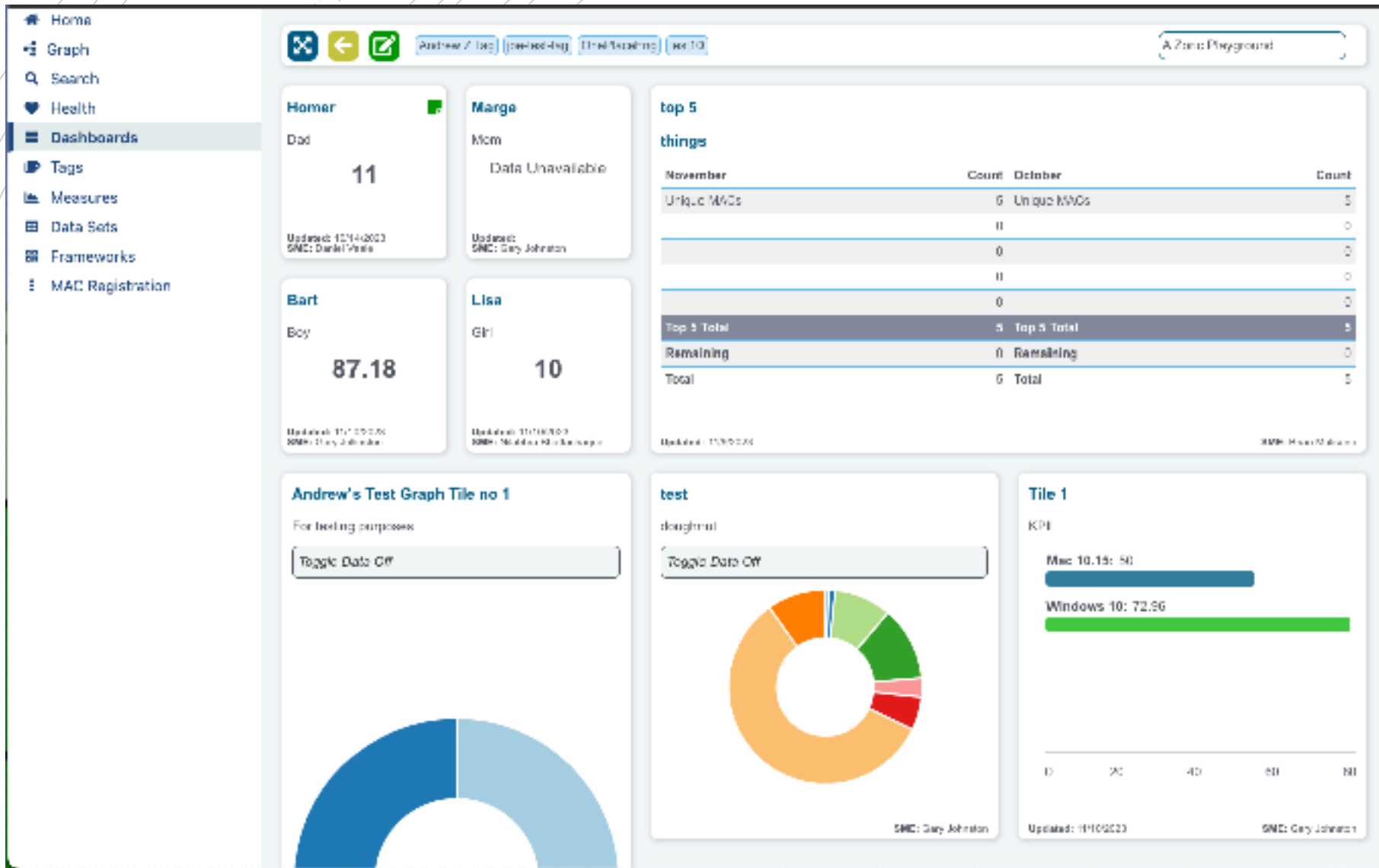
Located In HPF31ABF

Support: -oneplace-team-all

Version: 0.1.0

## Dark Mode

- The entire site had a Dark-Mode option. This was managed by a user system setting. The dark mode used high contrasting colors while maintaining the client's brand



## Configurable Dashboard

- Dashboards were configurable.
- Permitted users were able to create and change dashboards. This included.
- Dashboard “tiles” had multiple types, and could have different sizes.
- The entire dashboard was responsively designed so even smaller screens were usable.



## Configurable Dashboard

- Dashboards were configurable.
- Permitted users were able to create and change dashboards. This included.
- Dashboard “tiles” had multiple types, and could have different sizes.
- The entire dashboard was responsively designed so even smaller screens were usable.
- Dark mode shown

**Cargill OnePlace** Type  Name   HS

- Home
- Graph
- Search
- Health
- Dashboards
- Tags
- Measures
- Data Sets
- Frameworks
- MAC Registration

### MAC Address Registration

Register MAC Address | History

History						
Audit ID	Date	Property	Before	After	Changed By	
78832128	09/05/2023 08:45:01 PM	businessJustification	new one.	new one	8442118	
78832128	09/05/2023 08:45:01 PM	updatedOn	2023-09-08T20:27:00	2023-09-08T20:45:00	w142118	
78832688	09/05/2023 08:27:07 PM	businessJustification	new one	new one.	w142118	
78832688	09/05/2023 08:27:07 PM	updatedOn	2023-09-08T20:28:00	2023-09-08T20:37:00	w142118	
0	09/05/2023 08:26:00 PM	new	new	new	8442118	

Page 1 of 1 (5 results total)

Support: [oneplace-team-all](#)  
Version 0.1.0

## Searchable/ Sortable tables

- Table UI with ghosted search elements
- Once set active, or containing data the elements display fully. Options to filter in

The screenshot shows the Cargill OnePlace web application. The top navigation bar includes the Cargill logo, 'OnePlace', a search icon, and a user profile icon labeled 'HS'. A left-hand sidebar contains a menu with items: Home, Graph, Search, Health, Dashboards, Tags, Measures (highlighted), Data Sets, Frameworks, and MAC Registration. The main content area is titled 'Measures' and has several tabs: Details, Sources, Results, Links, Health, and Documentation (selected). Below the tabs, the 'Measure Name' is 'Andrew\_Test\_Measure'. A 'Markdown Help' checkbox is checked. The editor is split into two panes: a left pane for editing the markdown source and a right pane for the rendered output. The rendered output shows a heading 'Test Documentation' followed by a list item 'Test documentation for Andrew's test measure'. A 'Save' button is located at the bottom right of the editor area. In the bottom left corner, there is a footer with the text 'Support: -oneplace-team@all' and 'Version 0.1.0'.

## Documentation Editor

- Using Markdown Markup Language this editor was a two-way-bound code editor.
- Opening and closing the output window allowed the author to see the formatted output.
- A movable info box popup could be used with tips on markdown formatting.

The screenshot displays the Cargill OnePlace web application interface. On the left is a navigation sidebar with options: Home, Graph, Search, Health, Dashboards, Tags, Measures (highlighted), Data Sets, Frameworks, and MAC Registration. The main content area is titled "Measures" and contains a configuration form for a measure named "andrew\_test\_measure". The form includes fields for "Add Tags" (with "OnePlaceLog" selected), "SME Contact" (8442118), and "Description" (Andrew's best measure). Below the form are sections for "Cached Datasets" (Available Datasets), "Used Datasets", and "Unused Datasets" (testdataset). At the bottom is a "Query" editor with a SQL query:

```
1 select 'andrew_test'
2 as title, 1
3 as DimensionOne, 2
4 as dimensionTwo, 3
5 as dimensionThree, 4
6 as dimensionFour, 5
7 as dimensionFive, GETUTCDATE()
8 as createdOn
9
10
11
```

## SQL Editor with Code Highlight

- SQL Editor using CodeMirror and React. This gave SQL authors the ability to write readable code directly in the portal tool.
- Additional custom features included highlighting datasets referenced in the code.

# Part 3

## **Examples: Wallboard**

A client-server messaging system

Tech Stack: Node, Express, CSS



# more changes for the sake of change

we can just blame brian for everything

This Afternoon: Mostly  
Sunny 67

Tonight: Partly Cloudy  
44

Wednesday: Mostly  
Sunny 71

Wednesday Night:  
Partly Cloudy then  
Slight Chance Rain  
Showers 50

Thursday: Showers And  
Thunderstorms Likely  
72

LAST UPDATED: 11/17/2017 10:56 PM

## Wall Board frontend

- An internal ComResouce messaging system running on multiple Raspberry Pi's
- News rotates through an array of stories as hexagons dance in the background representing the ComResouce Logo using the particles.js library.
- A weather API gives the local forecast.

News Feed   Global Variables

### News Feed

Item Name:

Item Headline:

Item Text blur:

---

Item Name:

Item Headline:

News Feed   Global Variables

### Global Variables

Current Location

Uses GPS coordinates, obtainable from [Google Maps](#) by clicking on the map. this will show you the logitude/latitude where you click.

---

News Duration

In milliseconds. ie: 1000 milliseconds equals 1 second.

## Wall Board backend

- Using a Node server, Express.js the server back-end has two tabs
- The main tab is the newsfeed. Headline/blurb items that rotate through the client.
- Global variables. Sets the location for current weather and the duration each news item at rest before transitioning to the next.

# Part 4

**Examples: Wireframes**  
UX Designing for other developers

Tech Stack: HTML, JavaScript

