1

1a

Content of "Product Info" tab on deviceInfo screen
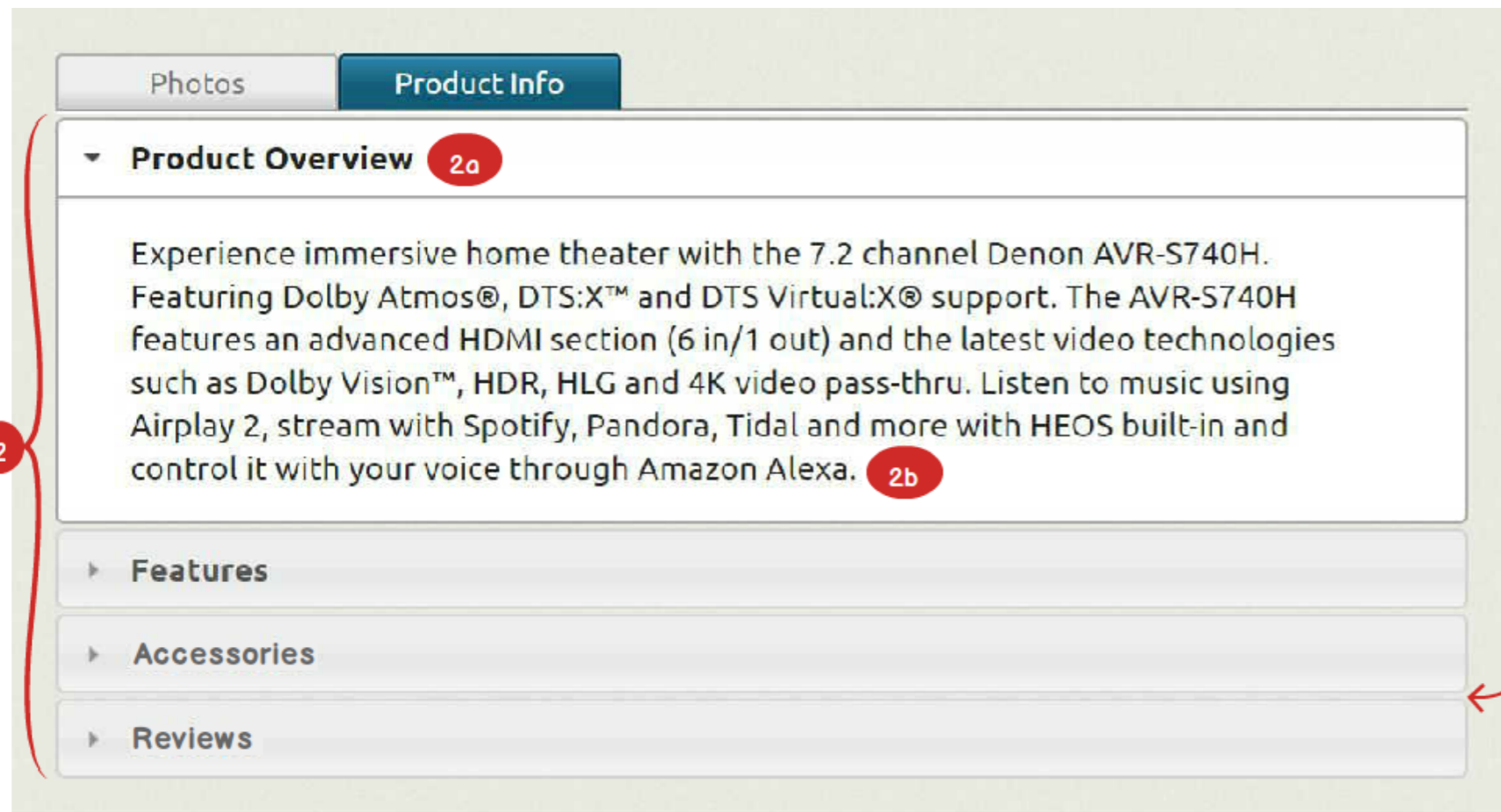
Overview of this document
We will be updating the accordion sections under the Product Info tab to display text from the ProdInfo.json response.
- Info layout/content described here is v1 - could change as we iterate/add more info.
- All Product Info tab data might need to be behind an internal system rule, feature flip, etc. since it CAN'T GO LIVE UNTIL we have final approval
- All layout, styling, & click behavior of Product Info tab/accordion is unchanged from exisiting functionality.

1 When user clicks Product Info tab, accordion loads (standard functionality)
- Unless user has already clicked Info button on previous page, this will be first time that ProdInfo call is being sent to manufacturer, so there may be a delay before .json file returns & accordion can populate.

1a. If there is a delay, page should show same spinner it currently does when tabs are clicked

Photos | Product Info

| Photos | **Product Info** |

▼ Product Overview ⬤2a

Experience immersive home theater with the 7.2 channel Denon AVR-S740H. Featuring Dolby Atmos®, DTS:X™ and DTS Virtual:X® support. The AVR-S740H features an advanced HDMI section (6 in/1 out) and the latest video technologies such as Dolby Vision™, HDR, HLG and 4K video pass-thru. Listen to music using Airplay 2, stream with Spotify, Pandora, Tidal and more with HEOS built-in and control it with your voice through Amazon Alexa. ⬤2b

▸ Features

▸ Accessories

▸ Reviews

⬤2   ⬤3

---

2. Content of Product Info tab can include up to 6 accordion sections. Sections should always display in this order:

Section title:	Section content
Product overview	ProdInfo.json > "overview"
Features	ProdInfo.json > "features" (Detailed on p.3)
Accessories	ProdInfo.json > "other_info" > "accessories"
Q&A	ProdInfo.json > "other_info" > "q_a"
Articles	ProdInfo.json > "other_info" > "articles"
Reviews	ProdInfo.json > "other_info" > "reviews"

2a. Each section's title should display exactly as listed above
2b. Each section's content should display exact values as returned by manufacturer, including any HTML returned.
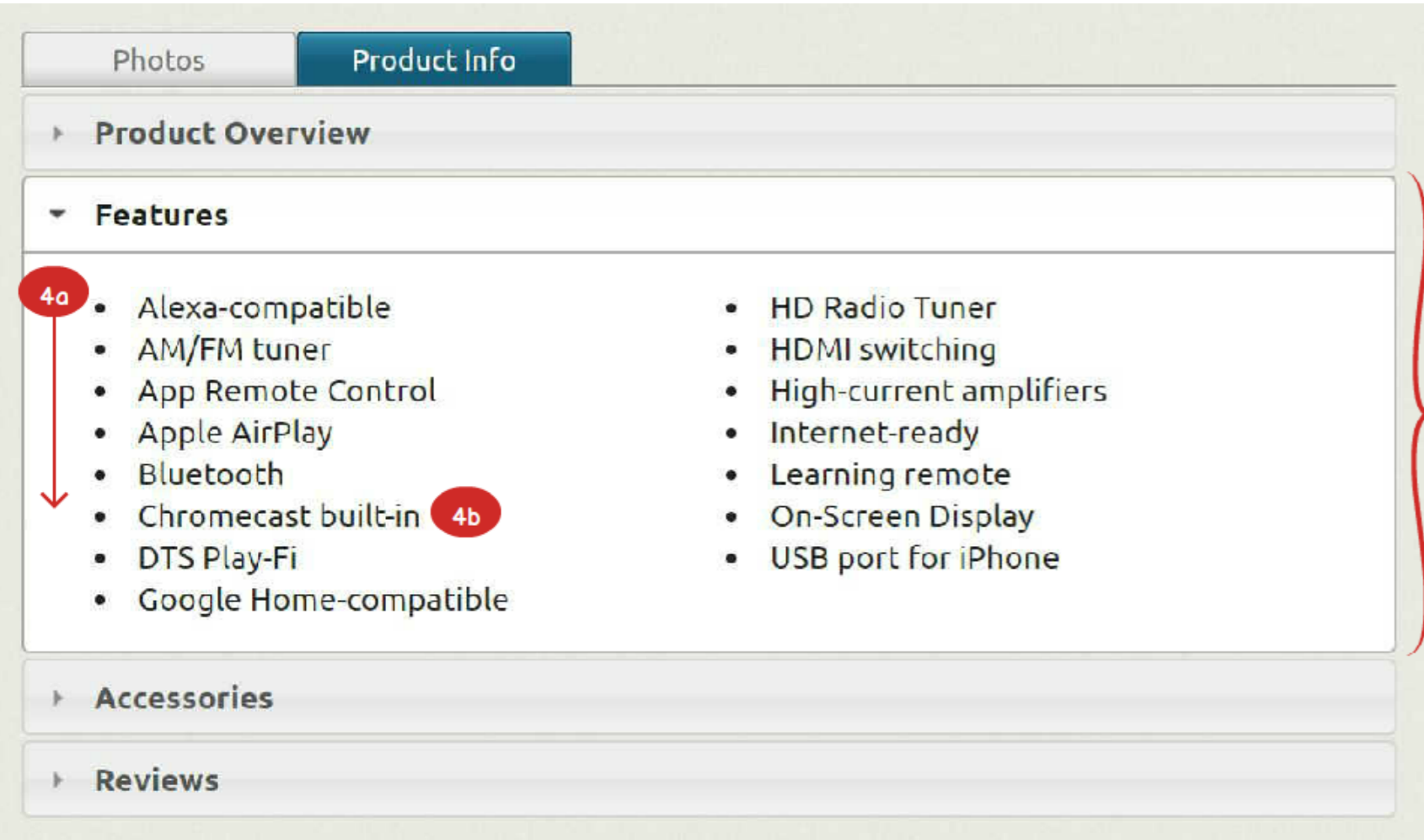- Devs may actually pull these values from our DB, but .json response locations are cited here for reference.

3. If any key/value pairs aren't returned, their accordion sections should not appear (in example shown here, "q_a" & "articles" did not return).

NOTE: the six accordion sections described above are only ones that can ever appear for a new product. Sections like "Compatibility", "Recommended Add-ons", and others that can currently appear for older products in the system are being phased out & should never be shown for new products.

As mentioned on p.1, click behavior of accordion is unchanged from exisiting page functionality. So:
- First accordion section (in this example, "Product Overview") should be expanded by defualt when user first clicks Product Info tab
- Clicking another tab expands it and collapses the one that was expanded previously

Product Info

| Photos | **Product Info** |

▶ Product Overview

▼ Features

4a (marker)
- Alexa-compatible
- AM/FM tuner
- App Remote Control
- Apple AirPlay
- Bluetooth
- Chromecast built-in **4b**
- DTS Play-Fi
- Google Home-compatible

- HD Radio Tuner
- HDMI switching
- High-current amplifiers
- Internet-ready
- Learning remote
- On-Screen Display
- USB port for iPhone

4 (marker)

▶ Accessories

▶ Reviews

```
"features": {
  "17": {
    "id": 17,
    "name": "Chromecast built-in"
  },
```

Features accordion section

4. "Features" section contains a list of product features, populated with values from from the ProdInfo.json response's 'features' object.
   - The 'features' object contains individual objects for each feature.
   - Each feature's object includes a 'name' value.

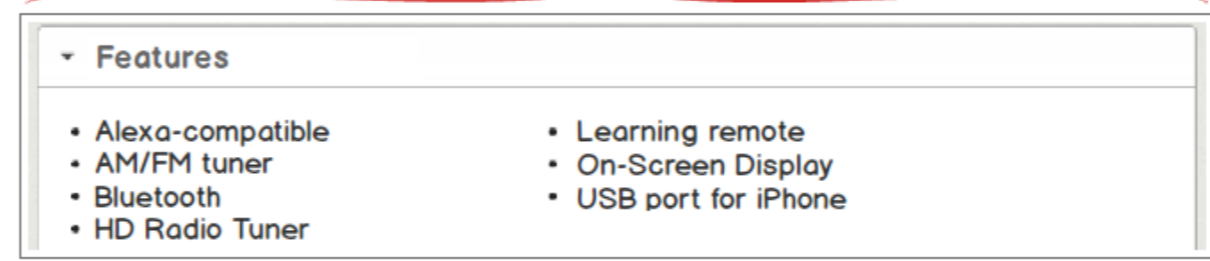4a. Features should be arranged in alphabetical order, using their 'name' values.

4b. Text seen on each line should be exact value of 'name' as returned for that feature.

Per discussion with Kate, columns should populate as follows:
   - If list contains two or more features, it will be in 2 columns (except on narrow screens, as discussed on p.4)
   - Length of each column should be calculated by taking total number of features returned & dividing by two (if odd number of features, put one more in left column)
   - Going in alphabetical order, features should "fill" the left column first; then the right column. So, this example 'features' return (containing 7 total features) would be laid out like this in the accordion section.
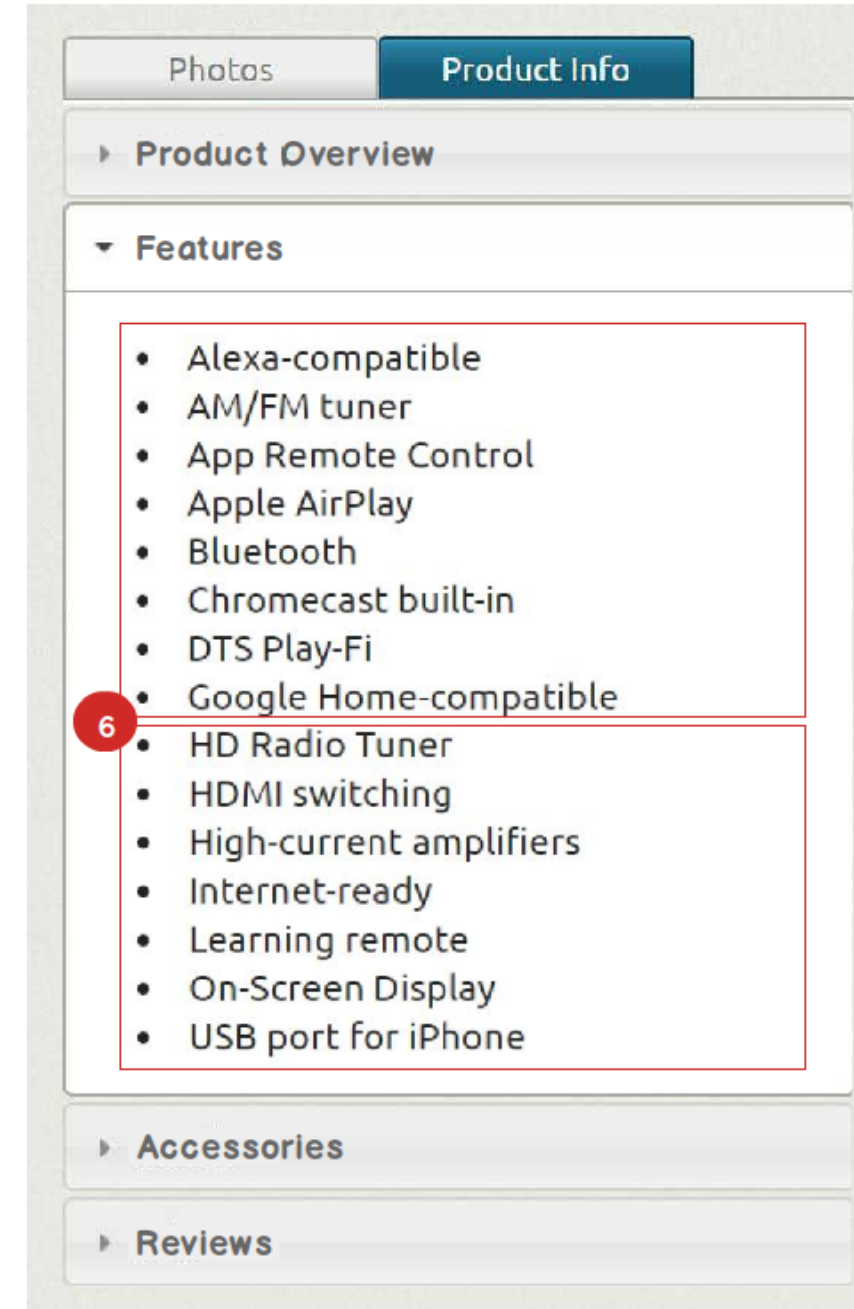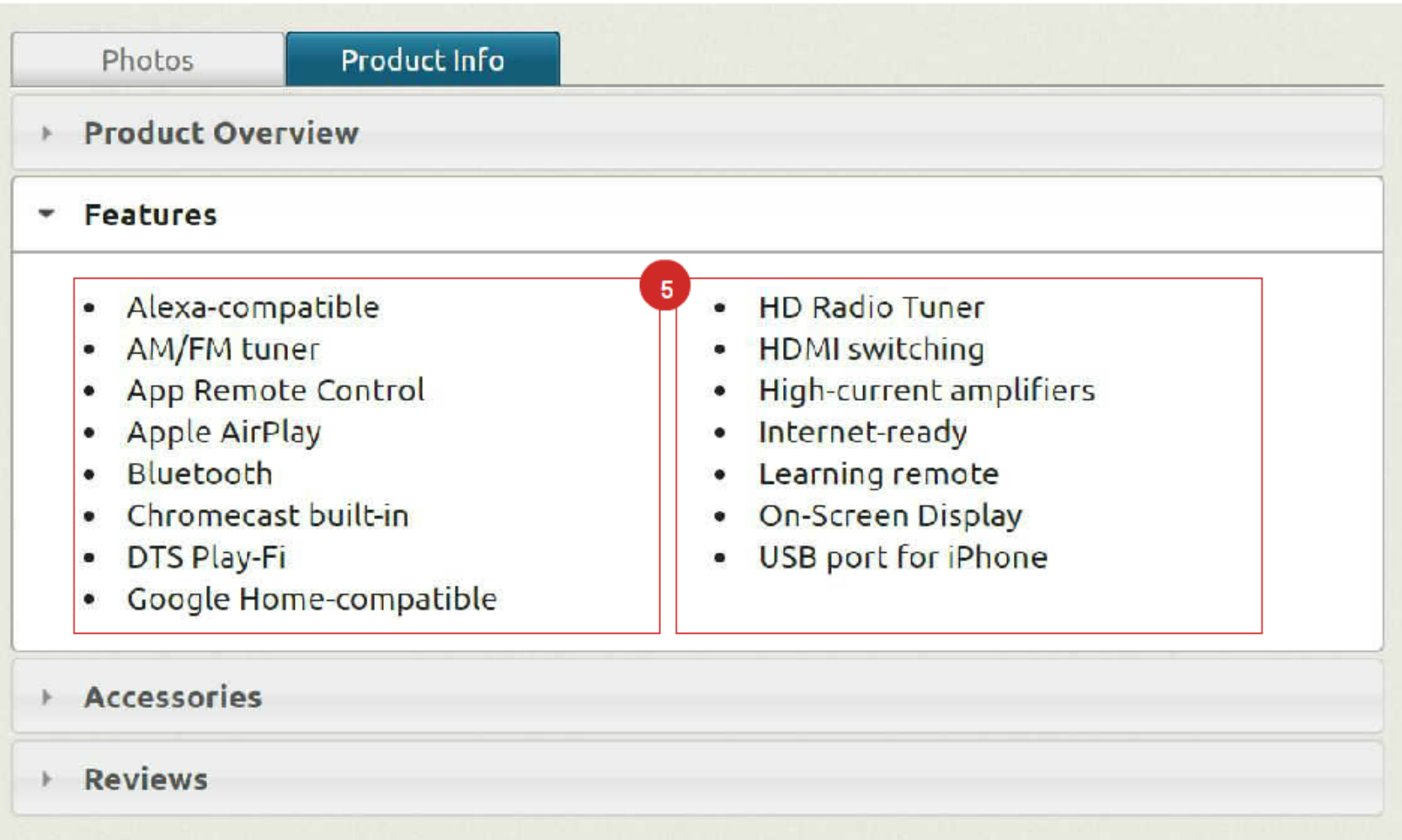
```
"features": {
  "41": {
    "id": 41,
    "name": "AM/FM tuner"
  },
  "52": {
    "id": 52,
    "name": "Alexa-compatible"
  },
  "54": {
    "id": 54,
    "name": "Learning remote"
  },
  "55": {
    "id": 55,
    "name": "HD Radio Tuner"
  },
  "324": {
    "id": 324,
    "name": "On-Screen Display"
  },
  "369": {
    "id": 369,
    "name": "USB port for iPhone"
  },
  "371": {
    "id": 371,
    "name": "Bluetooth"
  },
```

▼ Features

- Alexa-compatible
- AM/FM tuner
- Bluetooth
- HD Radio Tuner

- Learning remote
- On-Screen Display
- USB port for iPhone

Wide screen

Photos | **Product Info**

▸ **Product Overview**

▾ **Features**

(5)

- Alexa-compatible
- AM/FM tuner
- App Remote Control
- Apple AirPlay
- Bluetooth
- Chromecast built-in
- DTS Play-Fi
- Google Home-compatible

- HD Radio Tuner
- HDMI switching
- High-current amplifiers
- Internet-ready
- Learning remote
- On-Screen Display
- USB port for iPhone

▸ **Accessories**

▸ **Reviews**

Narrow screen

Photos | **Product Info**

▸ **Product Overview**

▾ **Features**

- Alexa-compatible
- AM/FM tuner
- App Remote Control
- Apple AirPlay
- Bluetooth
- Chromecast built-in
- DTS Play-Fi
- Google Home-compatible

(6)

- HD Radio Tuner
- HDMI switching
- High-current amplifiers
- Internet-ready
- Learning remote
- On-Screen Display
- USB port for iPhone

▸ **Accessories**

▸ **Reviews**

Features accordion section - responsive behavior on narrow screens

5. As agreed with Kate, each column of features is a flex item
   · Each column should have a min-width of 250px.
   · Columns should be separated by padding of 21px/1.5em
   · Columns should be equal width at various screen sizes (probably using flex-grow 1)
   · Columns will likely each use a <ul> element, with list-style-position: inside
   · Devs reach out to Kyle with any questions implementing this

6. When screen gets narrow enough (forcing Product Info tab narrow enough) that columns reach min-width, they should stack, with left column on top.