Bonik React Ecommerce Template

- ## Introduction

  Bonik is a multi vendor ecommerce theme built on React Next.Js aiming at faster performance, high code quality, SEO etc.

- ## Installation

  Open the command window in the root directory of this project. Type *npm install* for windows or *yarn* for mac and hit enter.

  To run this project locally run *npm run dev* in the command window. Project will be hosted on *localhost:3000*.

  Run *npm run build* for creating static files which will be found in the *build* folder of the project's root directory.

- ## Packages used

  1. React.Js
  2. Next.Js
  3. Styled Component
  4. Styled System
  5. Pure React Carousel
  6. Chart.Js
  7. Formik
  8. Typescript

- **Folder Structure**

  – root
  - – src
    - – components
    - – page-sections
    - – contexts
    - – data
    - – hooks
    - – interfaces
    - – models
    - – pages
    - – reducers
    - – utils

- **Theme**

  1. Global Styles can be changed from *./src/utils/globalStyles.ts* file
  2. Color can be changed from *./src/utils/themeColors.ts* file
  3. Color can be changed from *./src/utils/themeColors.ts* file
  4. Layout Size can be changed from *./src/utils/constants.ts* file

- **Selecting homepage**

  If you want to change the home page or root page, for example you want to use *grocery-1.tsx* as your main homepage, delete the existing *index.tsx* and rename *grocery-1.tsx* to *index.tsx* in the */src/pages* directory.

  Layouts are added at the bottom of the page. For example, in the following code we have added AppLayout at the bottom.

```
const IndexPage = () => {
  return (
    <div>
      <Section1 />
    </div>
  );
};

IndexPage.layout = AppLayout;

export default IndexPage;
```

● Data fetching

Each demo page gets its data by using getStaticProps function;

For example *market-1.tsx* fetching data in its *getStaticProps* function. This function uses REST api to get data.
HTTP calls are made in *utils/__api__/market-1* file. axios is used for calling the api, you can use any library you want.

```
// utils/__api__

const recentPurchase = async () => {
  const response = await axios.get('/api/recent-purchase')
  return response.data
}
```

● Mock server
    ● We used a mock server for demo purposes. You need to remove this from the template.
    ● Remove the *import "../__server__";* from *pages/_app.tsx* file

- Before removing the fake server, you need to create your server. Otherwise the will not work as they will not get data from the api.
- You can take a look at the models folder to get idea of data structure
- Also fake the server will give you the data structure. *src/__server__/index.ts*

- Fake API Example

```
// src/__server__/__db__/market-2/index.ts

// get all categories
Mock.onGet('/api/market-2/category').reply(() => {
  try {
    return [200, db.categories]
  } catch (err) {
    console.error(err)
    return [500, { message: 'Internal server error' }]
  }
})
```

- Components

1. Avatar

Props	Default Props	Description/Type
src		Image source/url (string)
size	48	Square size (number)
children		Children like number or text can be passed to show inside

		circle instead of image
*		All other image properties, ColorProps, SpaceProps, BorderProps, LayoutProps

```
<Avatar
  className="avatar"
  src="/assets/images/faces/7.png"
  size={64}
/>
```

2. Button

Props	Default Props	Description/Type
color		"primary"	"secondary"	"warn"	"error"	"inherit"
size	small	"small"	"medium"	"large"	"none"
borderRadius	5	number
variant		"text"	"outlined"	"contained"
fullwidth		Defines if button width will be 100% (boolean)
children		any
*		ColorProps, SpaceProps, BorderProps, LayoutProps, ShadowProps

```
<Button
  className="button-link"
  variant="contained"
  color="primary"
  p="1rem 1.5rem"
>
  Visit Collections
</Button>
```

3. IconButton / Round Button

Props	Default Props	Description/Type
color		"primary" | "secondary" | "warn" | "error" | "inherit"
size	small	"small" | "medium" | "large" | "none"
variant		"text" | "outlined" | "contained"
children		Icon
*		ColorProps, SpaceProps, BorderProps, LayoutProps, ShadowProps

```
<IconButton
  padding="4px"
  ml="12px"
  mt="-3px"
  size="small"
>
  <Icon size="1.5rem">close</Icon>
</IconButton>
```

4. Carousel

Props	Default Props	Description/Type
totalSlides	10	Number of total slides (number)
visibleSlides	5	Number of visible slides (number)
step	1	Number of visible steps(number)
autoPlay	false	boolean
interval	2000	number

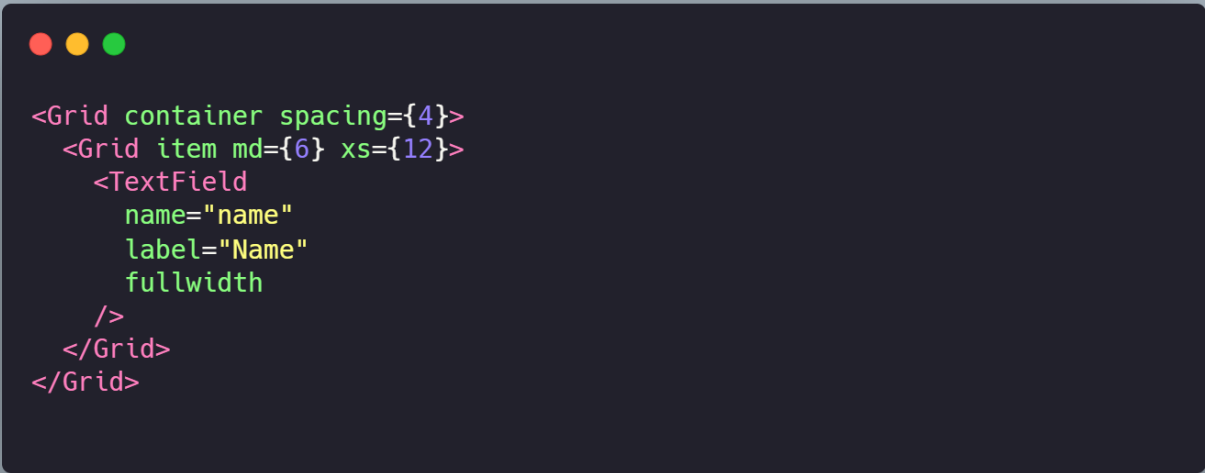| showDots	false	When true, dots will be shown below sliders (boolean)
showArrow	true	When true, arrow button will be shown on both side of the carousel (boolean)
dotGroupMarginTop	2rem	Defines top margin of Dots (string)
dotClass		Add class to dot group (string)
spacing	1.5rem	Defines space between each slide (string)
arrowButtonColor	secondary	"primary" | "secondary" | "warn" | "error" | "inherit"
leftButtonStyle		Add styles to left arrow button (object)
rightButtonStyle		Add styles to right arrow button (object)
children		ReactElement
*		All other props of pure-react-carousel

```
<Carousel
  totalSlides={5}
  visibleSlides={1}
  infinite={true}
  autoPlay={true}
  showDots={true}
  showArrow={false}
  spacing="0px"
>
  <CarouselCard1 />
</Carousel>
```

5. Grid

Props	Default Props	Description/Type
container		This will be defined as parent grid if it is true (boolean)
item		This will be defined as child grid if it is true (boolean)
containerHeight	unset	Set parent grid height (string)

xl		Defines number of columns on screen size above 1440px (number)
lg		Defines number of columns on screen size above 1024px (number)
md		Defines number of columns on screen size above 768px (number)
sm		Defines number of columns on screen size above 425px (number)
xs		Defines number of columns on screen size below 425px (number)
spacing	0	Defines space between each child grid (number). This number will be multiplied by 4px.
horizontal_spacing		Defines horizontal space between each child grid (number)
vertical_spacing		Defines vertical space between each child grid (number)
children		Grid	ReactElement

```jsx
<Grid container spacing={4}>
  <Grid item md={6} xs={12}>
    <TextField
      name="name"
      label="Name"
      fullwidth
    />
  </Grid>
</Grid>
```

6. Hidden

Props	Default Props	Description/Type
down		Hide children inside it when screen size is equal or below this amount (number/string)

up		Hide children inside it when screen size is above this amount (number/string)
children		any

```
<Hidden down={768} mr="1.75rem">
  <p>test</p>
</Hidden>
```

7. Icon

To use any svg file as an icon, put the svg file in */public/assets/images/icons* folder and use the same name in the code. For example, if you have a svg file called menu.svg then put this file in the above folder and in the code use *<Icon>menu</Icon>*

Props	Default Props	Description/Type
size		string
children		string
color		"primary" | "secondary" | "warn" | "error" | "inherit"
variant	medium	"small" | "medium" | "large"
defaultcolor		"currentColor" sets inherited color from parent "auto" uses original color of the svg file
*		SpaceProps

```
<Icon
  className="caret-icon"
  variant="small"
  defaultcolor="currentColor"
>
  chevron-right
</Icon>
```

8. Modal

Props	Default Props	Description/Type
open		Modal is open when true and vice-versa (boolean)
children		any
onClose		Function that fires on modal close

```
const [open, setOpen] = useState(false);

const toggleDialog = useCallback(() => {
  setOpen((open) => !open);
}, []);

<Modal open={open} onClose={toggleDialog}>
  <Card>this is a dialog</Card>
</Modal>
```

9. NavLink

Props	Default Props	Description/Type
href		Url of the link (string)
className		string
children		ReactElement
*		ColorProps, SpaceProps

```
<NavLink className="link" href="/help">
  Need Help?
</NavLink>
```

10. Pagination

Props	Default Props	Description/Type
pageCount		Number of pages (number)
onChange		(data: { selected: number }) => void
*		All props of react-paginate, SpaceProps

```
<Pagination
  pageCount={5}
  onChange={(data) => {
    console.log(data.selected);
  }}
/>
```

11. LinearProgress

Props	Default Props	Description/Type
variant	determinate	"determinate"	"indeterminate"
color	primary	"primary"	"secondary"	"warn"	"error"	"inherit"
value	75	Value with respect to 100 (number)
thickness	6	Thickness (height) of the progressbar (number)

```
<LinearProgress
  variant="determinate"
  color="primary"
  value={75}
  />
```

12. Radio

Props	Default Props	Description/Type
color	primary	"primary" | "secondary" | "warn" | "error" | "inherit"
labelColor		"primary" | "secondary" | "warn" | "error" | "inherit"
labelPlaceme nt		"start" | "end"
label		string | React.ReactChild
*		SpaceProps

```
<Radio
  name="cod"
  color="secondary"
  checked={paymentMethod === "cod"}
  label={
    <Typography ml="6px" fontWeight="600" fontSize="18px">
      Cash On Delivery
    </Typography>
  }
  onChange={e=> console.log(e)}
/>
```

13. RadioGroup

Props	Default Props	Description/Type
name		string
value		number | string
children		Radio
onChange		Event => void
*		SpaceProps

```jsx
    <RadioGroup
      name="group1"
      value={value}
      my={3}
      onChange={(e) => {
        setvalue(e.target.value);
      }}
    >
      <Radio py="0.5rem" value="a" color="primary" label="Radio Stat"
  />
      <Radio py="0.5rem" value="b" color="secondary" label="Radio Stat"
  />
      <Radio py="0.5rem" value="c" color="error" label="Radio Stat" />
      <Radio py="0.5rem" value="d" color="primary" label="Radio Stat"
  />
    </RadioGroup>
```

14. Rating

Props	Default Props	Description/Type
value	0	number
color	primary	"primary" | "secondary" | "warn" | "error" | "inherit"
outof	5	Total or highest value (number)
readonly	true	boolean
className		string

style		object
size		"small" | "medium" | "large"
onChange		(value: number) => void

```
<Rating
  outof={5}
  color="warn"
  size="medium"
  readonly={false}
  value={5}
  onChange={(value) => console.log(value)}
/>
```

15. Sidenav

Props	Default Props	Description/Type
position	right	"left" | "right"
width	200	number
open	false	Defines if the sidenav is open or not (boolean)
scroll	false	Defines if it is scrollable or not (boolean)
handle		Sidenav is toggled when clicking on handle (ReactElement)

```
<Sidenav
  position="left"
  scroll={true}
  handle={
    <IconButton size="small">
      <Icon>options</Icon>
    </IconButton>
  }
>
  <ProductFilterCard />
</Sidenav>
```

16. Step

Props	Default Props	Description/Type
title		string
disabled		boolean

17. Stepper

Props	Default Props	Description/Type
selectedStep	1	number
stepperList		Array of Step

```
const stepperList = [
  {
    title: "Cart",
    disabled: false,
  },
];

<Stepper
  stepperList={stepperList}
  selectedStep={selectedStep}
  onChange={(step,ind)=>{}}
/>
```

18. Sticky

Props	Default Props	Description/Type
fixedOn		Children's position will be fixed on this position (number)
children		ReactElement
onSticky		(isfFixed: boolean) => void
		Use ."section-after-sticky" class in the element next to the sticky component to prevent jumping of the UI.

```
<Sticky fixedOn={0} onSticky={isfixed=> console.log(isfixed)}>
  <Box>sticky header</Box>
</Sticky>
```

19. TextField

Props	Default Props	Description/Type
labelColor		"primary" | "secondary" | "warn" | "error" | "inherit"
label		string
errorText		string
fullwidth		When true, width will be 100% (boolean)
endAdornment		Element that will be shown at the end of the textfield (any)
*		SpaceProps

```
<TextField
  name="shipping_name"
  label="Full Name"
  fullwidth
  mb="1rem"
  onBlur={handleBlur}
  onChange={handleChange}
  value={values.shipping_name || ''}
  errorText={touched.shipping_name && errors.shipping_name}
/>
```

20. TextArea

Props	Default Props	Description/Type
labelColor		"primary" | "secondary" | "warn" | "error" | "inherit"
label		string
errorText		string
fullwidth		When true, width will be 100% (boolean)
*		SpaceProps

```
<TextArea
  name="comment"
  placeholder="Write a review here..."
  fullwidth
  rows={8}
  onBlur={handleBlur}
  onChange={handleChange}
  value={values.comment || ''}
  errorText={touched.comment && errors.comment}
/>
```

21. Box

Props	Default Props	Description/Type
*		ColorProps, SpaceProps, BorderProps, LayoutProps, PositionProps,FlexboxProps

```
<Box py="5rem" overflow="hidden">
  this is a box
</Box>
```

22. FlexBox

Props	Default Props	Description/Type
*		ColorProps, SpaceProps, BorderProps, LayoutProps, PositionProps,FlexboxProps

```
<FlexBox justifyContent="space-between" alignItems="center" mb="1.5rem">
  this is a flexbox
</FlexBox>
```

23. Card

Props	Default Props	Description/Type
boxShadow	small	"small" | "regular" | "large" | "badge" | "border" | "none"
hoverEffect	false	If true then card will larger shadow on hover (boolean)
*		ColorProps, SpaceProps, BorderProps, LayoutProps, PositionProps,FlexboxProps

```
<Card p="40px 72px" boxShadow="small" bg="primary.main">
  this is a card
</Card>
```

24. CheckBox

Props	Default Props	Description/Type
color	secondary	"primary" | "secondary" | "warn" | "error" | "inherit"
labelColor		"primary" | "secondary" | "warn" | "error" | "inherit"
labelPlacement		"start" | "end"
label		any
size	18	Defines height/width of the checkbox (number)

*		ColorProps, SpaceProps, BorderProps, LayoutProps, PositionProps,FlexboxProps

```
<CheckBox
  name='any-name'
  value='any-value'
  color="secondary"
  label={<SemiSpan color="inherit">{item}</SemiSpan>}
  my="10px"
  onChange={(e) => {
    console.log(e.target.value, e.target.checked);
  }}
/>
```

25. Chip

Props	Default Props	Description/Type
cursor		Cursor types like pointer, move etc. (string)
boxShadow		string

```
<Chip
  position="absolute"
  bg="primary.main"
  color="primary.text"
  fontSize="10px"
  fontWeight="600"
  p="5px 10px"
  top="10px"
  left="10px"
>
  50% off
</Chip>
```

26. Container

Props	Default Props	Description/Type
*		ColorProps, SpaceProps, LayoutProps, PositionProps

```
<Container pb="1rem">This is a container</Container>
```

27. Divider

Props	Default Props	Description/Type
*		SpaceProps, LayoutProps

```
<Divider width="200px" mx="auto" />
```

28. DropZone

Props	Default Props	Description/Type
onChange		(files: []) => void

```
<DropZone
  onChange={(files) => {
    console.log(files)
  }}
/>
```

29. Image

Props	Default Props	Description/Type
*		SpaceProps, LayoutProps, BorderProps

```
<Image
  src="/assets/images/products/bglight.png"
  alt="fashion"
  height="52px"
  borderRadius={8}
/>
```

30. Menu

Props	Default Props	Description/Type
direction	left	"left"	"right"
handle		Clicking on this handle will show menu (boolean)
className		string
children		MenuItem or any

```
const languageList = [
  {
    title: "EN",
    imgUrl: "/assets/images/flags/usa.png",
  }
];

<Menu
  direction="right"
  handler={
    <FlexBox
      className="dropdown-handler"
      alignItems="center"
      height="40px"
      mr="1.25rem"
    >
      <Icon size="1rem">option</Icon>
    </FlexBox>
  }
>
  {languageList.map((item) => (
    <MenuItem key={item.title} onClick={handleLanguageClick(item)}>
      <Image
        src={item.imgUrl}
        borderRadius="2px"
        mr="0.5rem"
        alt={item.title}
      />
      <Small fontWeight="600">{item.title}</Small>
    </MenuItem>
  ))}
</Menu>
```

31. Select

Props	Default Props	Description/Type
value		any
defaultValue		any
label		string
errorText		string

options		Array of the following object
		{
		  label: any;
		  value: any;
		}

```
const countryList = [{ label: 'Afghanistan', value: 'AF' }]

<Select
  mb="1rem"
  label="Country"
  options={countryList}
  value={values.shipping_country || 'US'}
  onChange={(country) => {
    setFieldValue('shipping_country', country)
  }}
  errorText={touched.shipping_country && errors.shipping_country}
/>
```

32. TableRow

Props	Default Props	Description/Type
*		ColorProps, SpaceProps, BorderProps

```
<TableRow padding="0px 18px" boxShadow="none" bg="none">
  <div>table cell</div>
  <div>table cell</div>
  <div>table cell</div>
</TableRow>
```

33. Typography, H1, H2, H3, H4, H5, H6, Span, SemiSpan, Small, Paragraph, Tiny

Props	Default Props	Description/Type

ellipsis		Text will ellipse in single line when true (boolean)
*		ColorProps, SpaceProps, BorderProps, LayoutProps, FlexboxProps, TypographyProps

```
<Typography fontSize="14px" fontWeight="900">
   random typography
</Typography>
```

34. Categories

Props	Default Props	Description/Type
open		If true, the dropdown menu will be open forever and ignore clicks.
children		Clicking on children will toggle the dropdown menu if no props is passed for open

```
<Categories>
   <FlexBox color="text.hint" alignItems="center" ml="1rem">
      <Icon>categories</Icon>
      <Icon>arrow-down-filled</Icon>
   </FlexBox>
</Categories>
```