

TASK DESCRIPTION

Contents

Synopsys.....	1
Pages to make	2
Acceptance criteria	2
Markup.....	2
Functionality	2
Homepage (Start).....	2
Catalog	3
Item details page.....	3
Shopping bag.....	3
Extra miles.....	3
Item details page.....	4
Shopping bag.....	4
Carousel functionality	4
Filter functionality.....	4
Browser support	6
Tools.....	6
List of fonts.....	6
Recommended file structure	6

Synopsys

This **FINAL INDIVIDUAL** task combines all your knowledge and skills.

Every student will defend it to interviewers personally.

Your goal is to create minimalistic e-commerce website.

Implementation is expected as set of static HTML files with responsive layout with (RWD techniques, Fluid layout).

You should add necessary basic JS.

Usage of CSS/JS libraries/frameworks is disallowed (if there are no special approve) and penalty might be applied if you use one.

There is list of extra miles that are not required to implement, but will be a bonus when implemented.

Pages to make

- Homepage (1_start)
- Catalog (2_catalog)
- Item details page (3_item)
- Shopping bag (4_shopping_bag)

Acceptance criteria

Your solution would be evaluated by the following criteria:

Markup

- For screen width larger than 375px (including) pages should use mobile layout.
- For screen width larger than 768px (including) page should use tablet layout.
Page layout can be switched from mobile to tablet layout earlier than 768px if you see that this wouldn't break your layout.
- For screen width larger than 1024px page should use desktop layout.
- Main content area shouldn't stretch more than 1180px.
- Used HTML tags should be semantic.
- Semantic names of classes and ids.
- Correct using of CSS3 rules.
- Markup must be compatible with list of supported browsers.
- All links and buttons have noticeable hover and active state.
- Mobile version should have menu as shown in 5_menu file.
- Penalty might be applied to noticeable deviations from design. Do your best to make your layout as close as possible to design.

NOTE: There should be no horizontal scroll bar unless device width is less than 375px. In this case layout shouldn't be broken and horizontal scroll appears.

Functionality

List of requirements for basic functionality (navigation, user interaction)

Homepage (Start)

- Carousel (slider) at the top of Homepage should show static images.
Images must be changed every 10 seconds.
(**NOTE:** Animations for carousel are part of **Extra miles** section)
- Click on images in carousel should navigate user to Catalog or Item details page (in alternating order: first, third etc. – Catalog page; second, fourth etc. – Item details page).
- Previous/Next arrow in Carousel should be visible only on Desktop. Click should show previous and next image respectively; Reset image change timer (next image should be shown in 10 seconds, and not in next second if user clicked on button before scheduled image change).

- Number of dots in pager (right under carousel) should match number of images in carousel. Highlighted dot must match corresponding image.
- Click on “Nike Red” (block under carousel) should navigate user to Item details page.
- Click on “Caps & Hats” (block under carousel) should navigate user to Catalog page.
- Search field should be collapsed by default on tablet. Click on magnifying glass icon should expand search field to the left. (It should be displayed above menu items in header, see)

Catalog

- Hover over item block should highlight item title and show overlay with “View Item” text for item image.
- Click on item block should navigate user to Item details page.
- Filters for Catalog page should be implemented as dropdown lists on Desktop.
 - Dropdown should be opened on hover over filter bar element.
- Selected filter value should be highlighted in dropdown list.

NOTE: Filter change functionality and implementations for Mobile and Tablet are part of **Extra miles** section.

Item details page

- Click on “Back to catalog” link should navigate user to Catalog page.
- Click on buttons in Size and Color sections should highlight it. Only one button per section can be selected.
- Click on "Add to bag" item should update items count and total price in header. Items count should be increased by one, total price should be increased by an arbitrary value in the range from 250 to 400.

NOTE: Proper cart implementation is listed in **Extra miles** section.

Shopping bag

- Hover over item block should highlight item title and show overlay with “View Item” text for item image.
- Click on "Clear bag" link should remove all product items from Shopping bag and replace them with text "Your shopping bag is empty. Use Catalog to add new items".
- Click on "Buy now" should remove all product items from Shopping bag and replace them with text "Thank you for your purchase".
- In both cases listed above: remove total price from header and items count should be displayed as **(0)**.

Extra miles

This is list of bonus tasks that are not required to implement, but will give you a bonus if implemented.

Item details page

- Implement photos switcher
Click on thumbnail should replace main image with full size image of itself, and make thumbnail active.

Shopping bag

(full shopping bag functionality with JS)

- Create one more detailed item page (so you will be able to add different items to shopping bag).
- Create persistent storage for Shopping bag data so it is saved when user navigated across the site (use `LocalStorage`).
- Click on "Add to bag" button should add item to persistent shopping cart storage:
 - If two items in bag has same name, size and color - they should be combined on the Shopping bag page. Quantity field in shopping bag item should reflect number of similar items.
 - If two items in bag has same name, but different size and/or color - they shouldn't be combined.
- Every time when user click on "Add to bag" button - items count and total price in header should be recalculated.
- Every time when user click on "Remove item" - quantity for this item should be decreased by one.
 - Items count and total price (in header and on the shopping bag page) should be recalculated.
 - If quantity is equal to 1 and user click on "Remove item" - item should be totally removed from the Shopping bag.

You can create and use your own stub/mock data.

Carousel functionality

- Implement "fade in" animation for image change in carousel.
- Click on dot from carousel pager should show corresponding image and reset image change timer.
- Implement swipe support to change images on tablet and mobile devices.

Filter functionality

This extra mile describes behavior of filters bar and related element. Real filtering of items page is not supposed.

- Only one value per filter can be selected for simplicity.
- For tabled and mobile all filters are combined into single block which should be opened by click anywhere on filters bar and displayed above page content.
- Click on value in any category should highlight this value and update filters bar:
 - Desktop: If "Not selected" is selected value for filter category – only category title should be displayed in the filters bar element. Otherwise both category title and value should be displayed in filter bar element.

- Tablet and mobile: All selected filter values and category titles should be concatenated in single string with “,” as separator. If filter category has selected value it should be displayed and highlighted in filters bar. Otherwise filter category title should be displayed (without highlight).

Browser support

Your implementation should be compatible with latest versions of following browsers:

- Chrome
- Firefox
- Edge
- IE11

Tools

- Code editor of your choice
- Chrome DevTools
- Pixel Perfect (Chrome plugin)
- Adobe Photoshop
- http-server package (<https://www.npmjs.com/package/http-server> , Requires installed Node.js) or any other server that can server static html files to avoid problems with LocalStorage.

List of fonts

If you encounter proprietary fonts in PSD filed- use free alternatives from list below:

- Helvetica Neue → Roboto
- Futura PT → Open Sans
- Times → Crimson

Recommended file structure

Files and folders structure may vary depending on your implementation.

