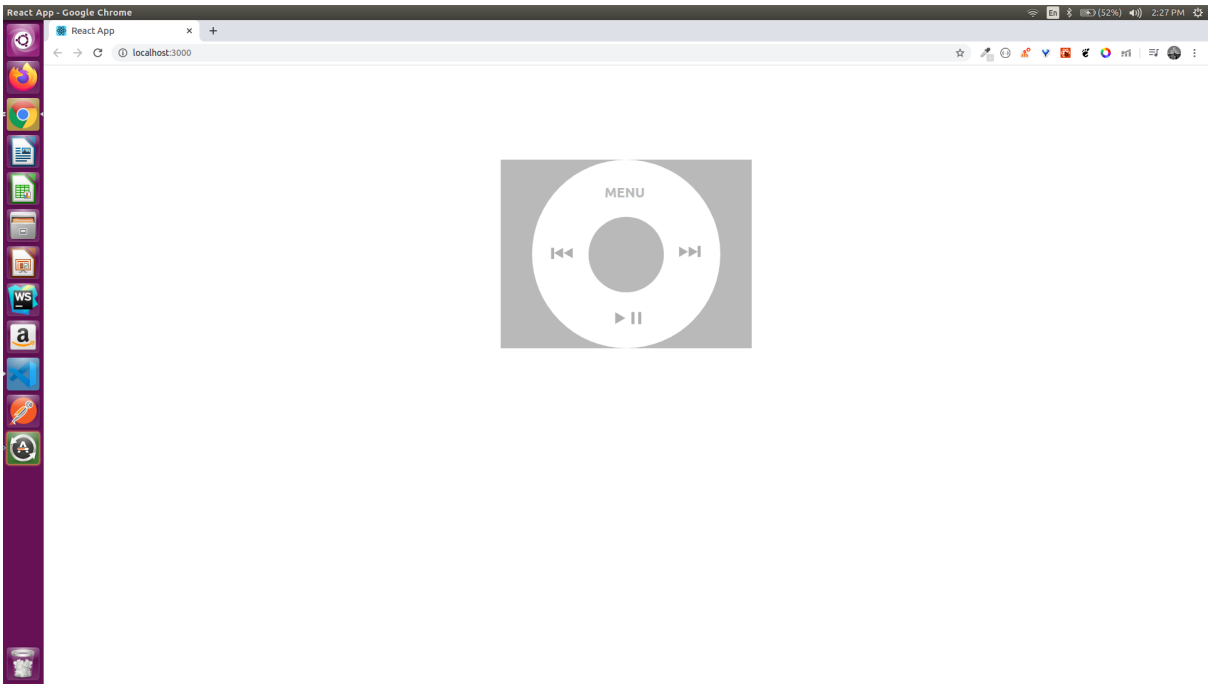
Steps

1. ## Design wheel UI

   Use CSS like position:absolute to position various **divs** on top of the wheel div. Final result should look something like this. You can improve CSS later.

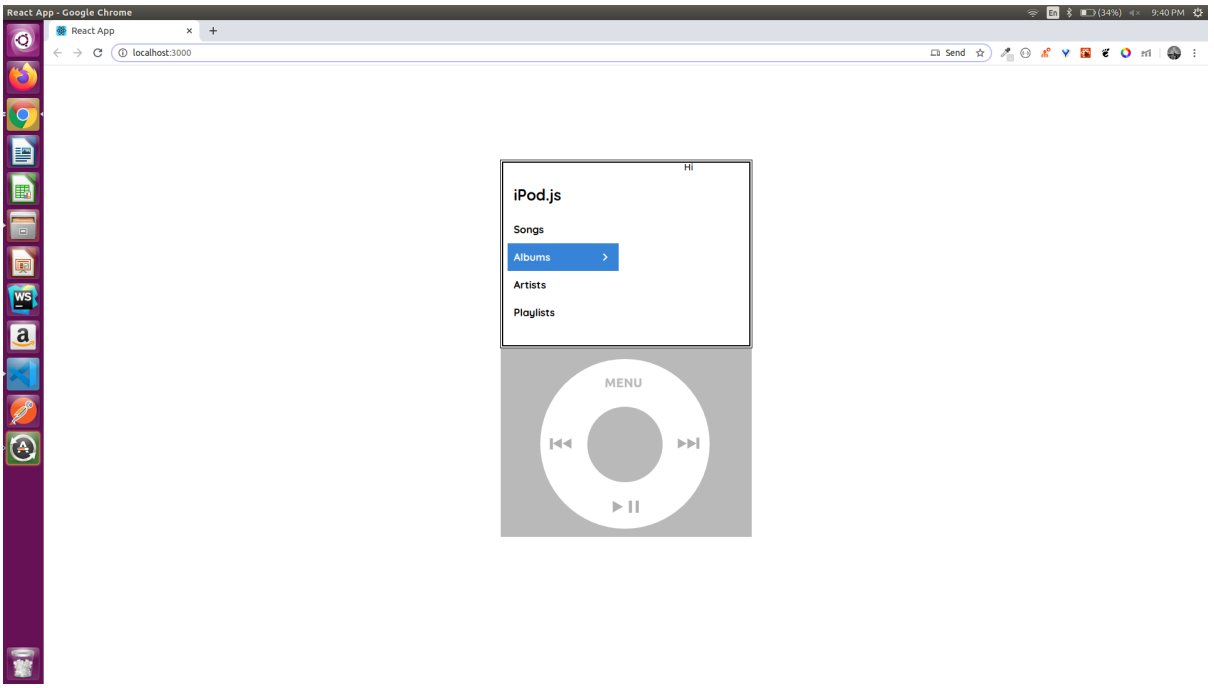


2. ## Add event listener to capture rotation around wheel

   Look up https://zingchart.github.io/zingtouch/ library. And see how you can use its rotate event to capture rotate events. Final result should look something like this. Log onto the console when you move around the wheel.

3. Add List

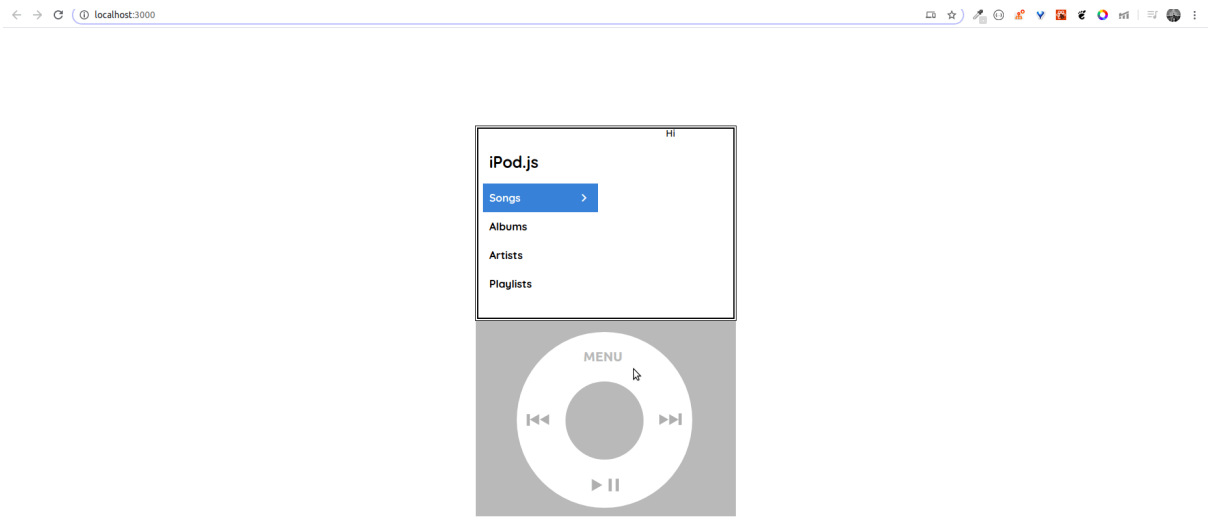
   Add a seperate screen component which will contain our side menu and display. Add an active class to one of the list items so it has a different background color. It will look something like this.



4. Change active menu item using wheel

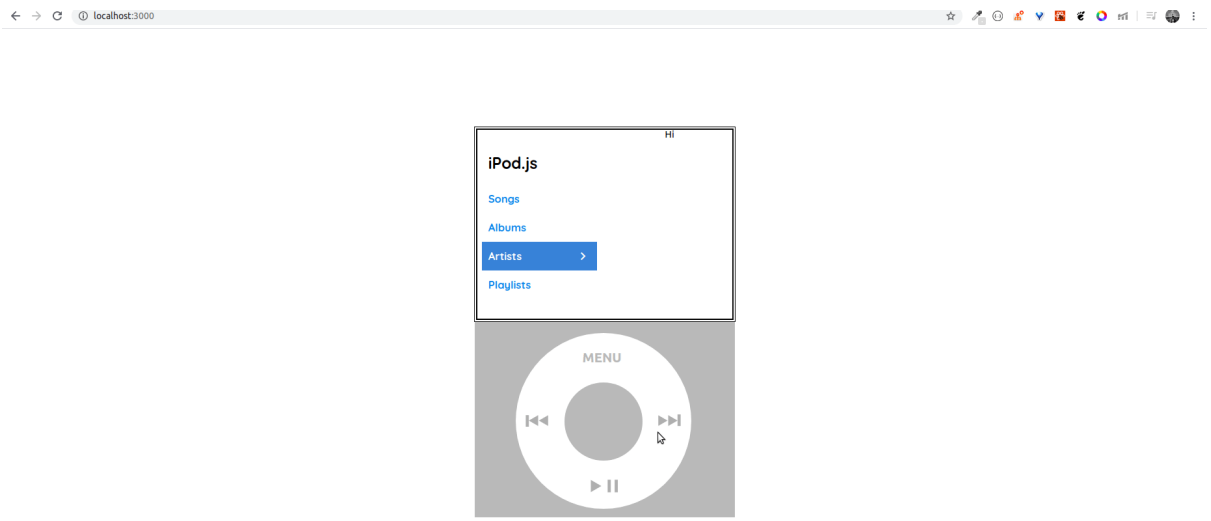
   Use rotate event (ZingTouch) you used to change the active menu item using **distanceFromLast** property attached to rotate event. Change the active item menu based on angle for ex. If angle changed>15 deg change the menu item. Final result will look something like this.

   Note: Add draggable:false to prevent drag on clicking and moving mouse pointer



5. Handling click on a menu item

   On clicking of menu hide the menu and show another component, depending upon the menu option the user clicked.



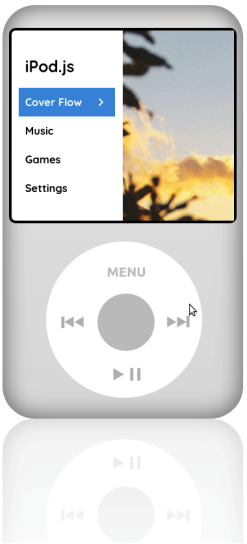
6. Add onClick on Menu button

   Add an event listener to the MENU button to go back one page. So from /songs it should go back to the home page.

7. Improve UI

   Now you can focus on css and improve the UI. And add different screens according to different routes.

   Finally it should look something like this


iPod.js
Cover Flow
Music
Games
Settings
MENU