



## **React boot camp**

# **TASKS**

---

### **Legal Notice:**

This document contains privileged and/or confidential information and may not be disclosed, distributed or reproduced without the prior written permission of EPAM®.

---

## CONTENTS

1	TASK 1 - (CORE CONCEPTS PT.1, PT.2) .....	3
2	TASK 2 - COMPONENTS, ROUTING .....	5
3	TASK 3 - FLUX, REDUX .....	11
4	TASK 4 - TESTING, PROJECT STRUCTURE.....	11

## 1 TASK 1 - (CORE CONCEPTS PT.1, PT.2)

You should only layout these pages using JSX. They will be used in future tasks.

Each category has its own list of tasks and shows only them, even if the category has nested categories.

Selected category should be highlighted. If no category is selected hide right section with tasks input.

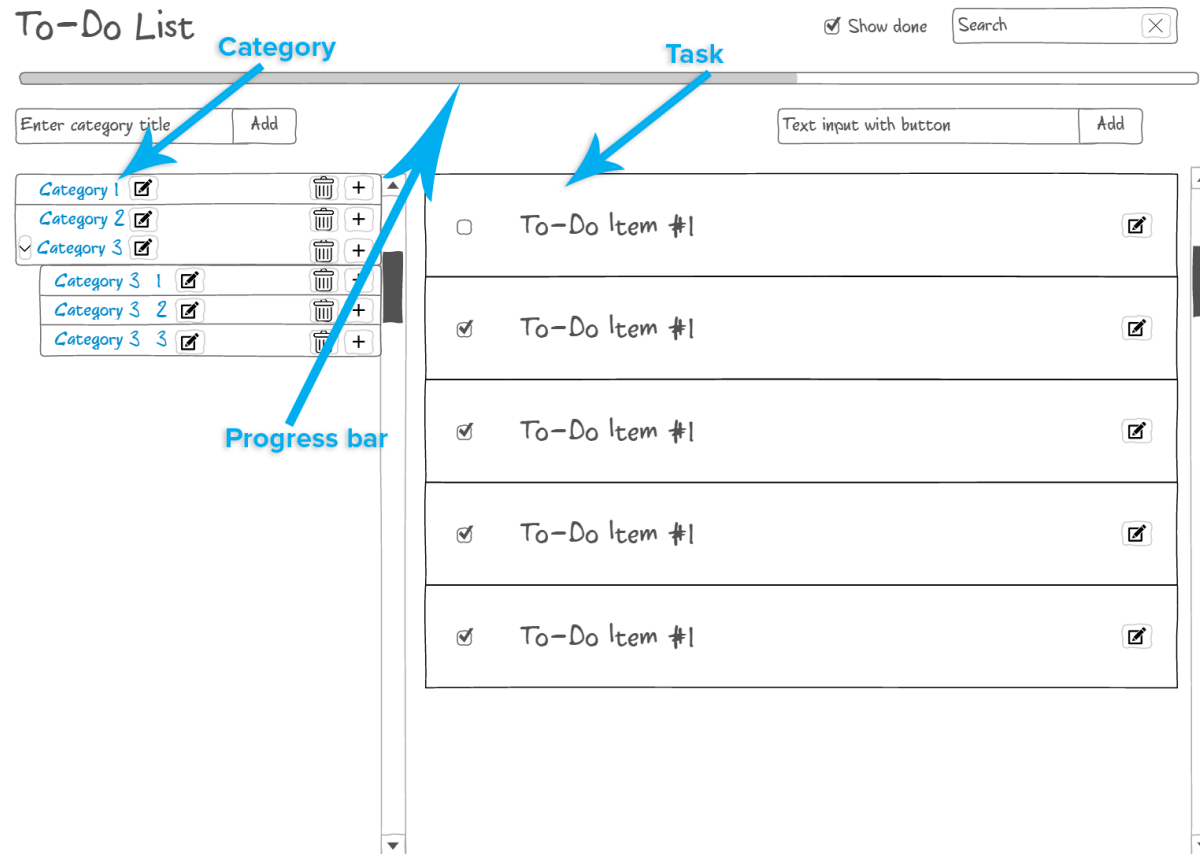
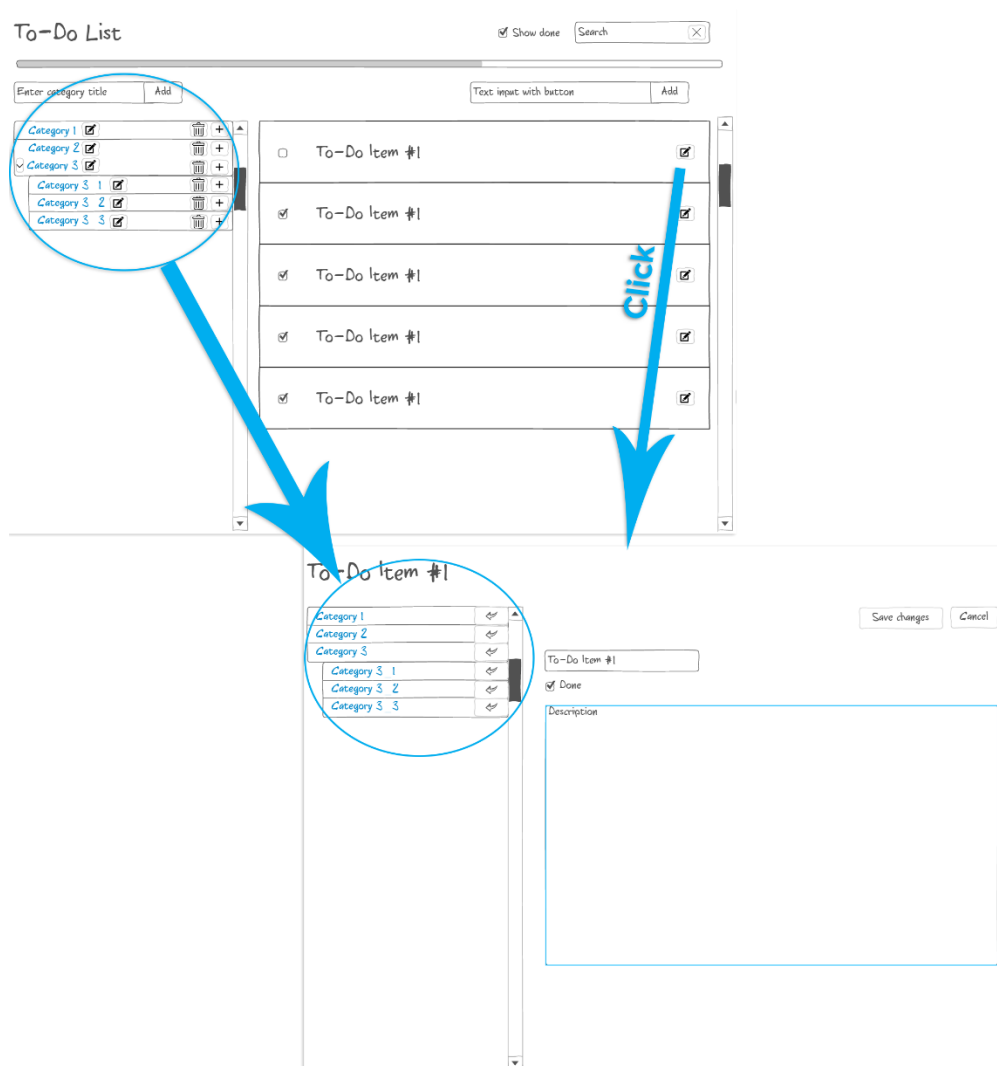


Figure 1 Main page

**Figure 2 Task edit page**

## 2 TASK 2 - COMPONENTS, ROUTING

Create such TODO application

Filter and selected category and task must be in URL. The search filter applies to tasks.

The screenshot shows a web application titled "To-Do List". At the top right, there is a "Filter" section with a checked "Show done" checkbox and a "Search" input field. Below this, there are two input fields: "Enter category title" with an "Add" button, and "Text input with button" with an "Add" button. On the left side, there is a "Categories tree" containing a list of categories: "Category 1", "Category 2", "Category 3" (which is expanded to show "Category 3 1", "Category 3 2", and "Category 3 3"). Each category has a trash icon and a plus icon. On the right side, there is a "Tasks list" containing a list of tasks. Each task has a checkbox, a text input field, and a trash icon. The tasks are: "To-Do Item #1" (unchecked), "To-Do Item #1" (checked), "To-Do Item #1" (checked), "To-Do Item #1" (checked), and "To-Do Item #1" (checked). Annotations include a blue circle around the "Categories tree" and a blue oval around the "Filter" section.

To-Do List

Filter

☒ Show done Search

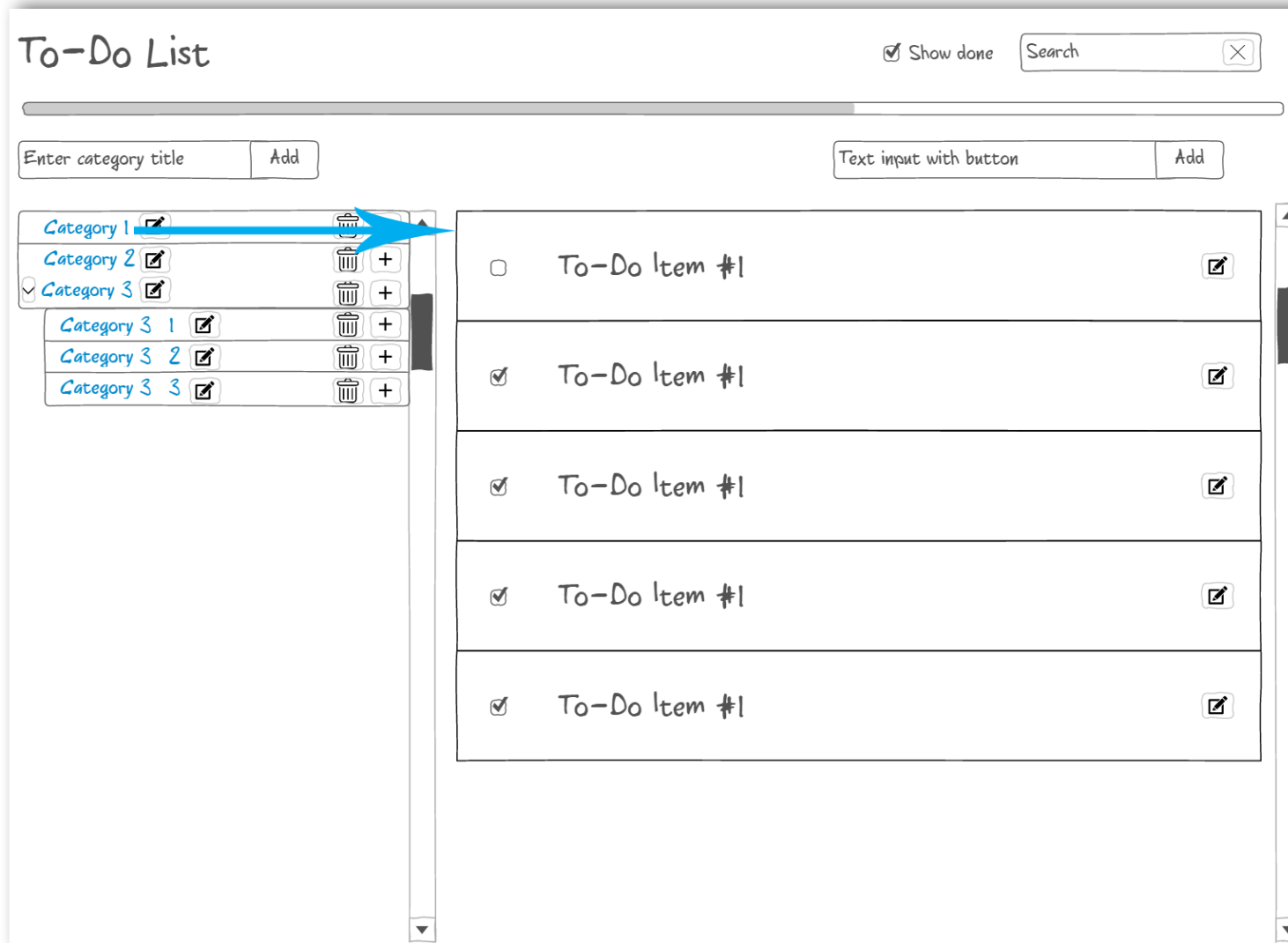
Enter category title Add

Text input with button Add

Categories tree

Tasks list

When you click on category you will see its tasks

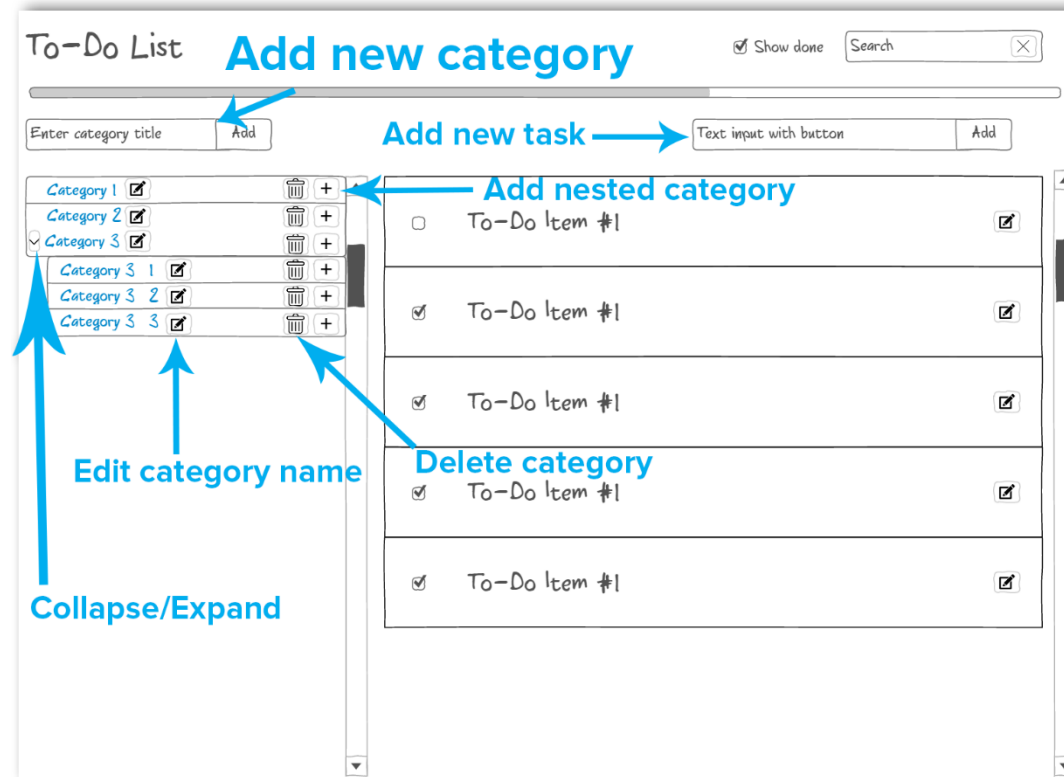


The categories tree may have infinite hierarchy

When user adds new category it goes to top of the tree. The UI of how to get the name is up to you (i.e. modal window or inline input)

When user deletes category all the nested categories are also become deleted. Confirmation should be shown.

New task goes to top of the list



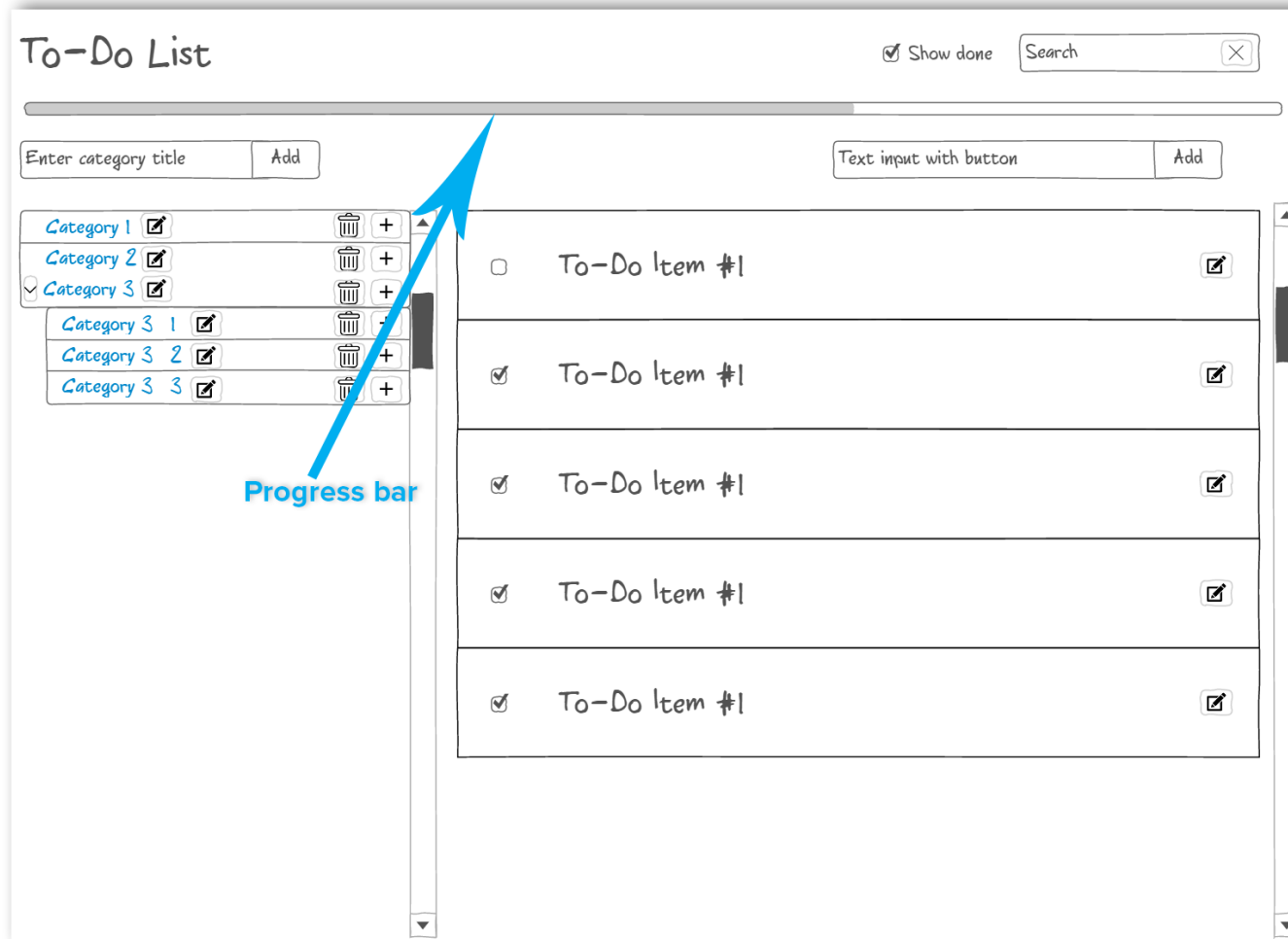
By clicking “Edit category” user can rename the category. The UI should be the same as in adding new category.

\* User can’t move categories, only delete or create new category. Also user can’t delete tasks.

Category flow:

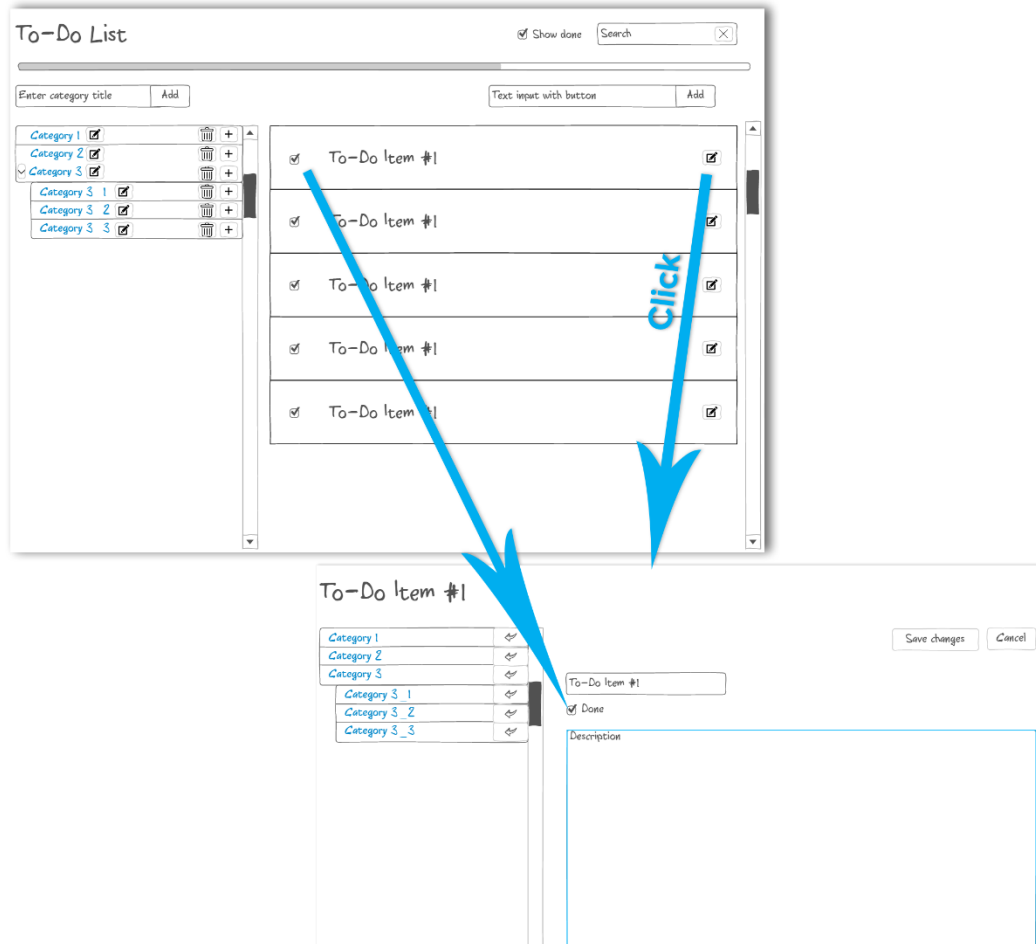
1. Case - you type category title in the input and click ‘Add’
2. Case - you click on ‘Add nested category’ and then i.e. modal window is appeared in which you can type name for nested category
3. Case - you click on ‘Edit category name’ and then the component of the second case is appeared with filled value. The purpose of it is to show reusable components approach.

Progress of all completed categories. Completed category is a category in which all tasks are done or if it's empty (empty categories are always visible). If parent category is completed but child is not, the parent category stays in the tree but the progress bar counts it as completed.



It is possible to edit particular task. It should be added to URL.





Also, you can move task into another category by clicking move icon. The move icon is hidden in the current task is presented in the category.

## To-Do Item #1


Category 1	↩
Category 2	↩
Category 3	↩
Category 3_1	↩
Category 3_2	↩
Category 3_3	↩

Save changes Cancel

To-Do Item #1

☒ Done

Description



**Move task into another category**

### **3 TASK 3 – FLUX, REDUX**

- Add Undo/Redo (You can add these buttons to any place you want in the application)
- Use Flux as your data flow container
- Use Redux as your data flow container

### **4 TASK 4 – TESTING, PROJECT STRUCTURE**

- Create tests special for Redux
- Reorganize and refactor your project

REVISION HISTORY					
Ver.	Description of Change	Author	Date	Approved	
				Name	Effective Date
1.3	Description update		28.11.2016		
1.4	Add info about adding new category		28.11.2016		
1.5	Add info about search and no category selection case		15.12.2016		
1.6	Removed E2E task		11.01.2017		