**ToDo App structure and flow.**

This app can run on any modern browser and let its user add todo items, view them any time later, mark them done, undo done tasks and delete done tasks.

The app is designed using MVC pattern.

ToDoModel.js – This is the model of the app where the data is stored.Is responsibility is just to store the data and not worry about rendering or fetching and saving the data.

ToDoView.js – This is the view part where the main rendering logic stands. It uses the model data to render the view .This does not worry about data or events over the view. When ever a render event it triggered this just retenders the model data on page. (in other way we can use some template engine like handlebars ,JADE ,etc to take care of this.).

ToDoController.js – This is the main controller, which binds the model to the view. Takes care of any event handling over the page. It does not directly manipulate the view but manages the triggering of render event on change of model data. This is also responsible for fetching and saving the data from local storage.

In real app , there will be some ajax request to populate the model data or it will be rendered from server side.

ToDoApp.js – this is the starting point of the application. This checks of the app already exists in local storage or not and creates it if not with some fake data. Actually, it will get the data first time via ajax or some ting which will be stored in local storage. This then triggers the controller on window load to start the app.

ToDoUtil,js – This is separate module which holds common functionality like binding events, getting elements using ID ,setting and getting data from local storage. Its instance is created and set in to window so any one can use it. (Something like jQuery ).