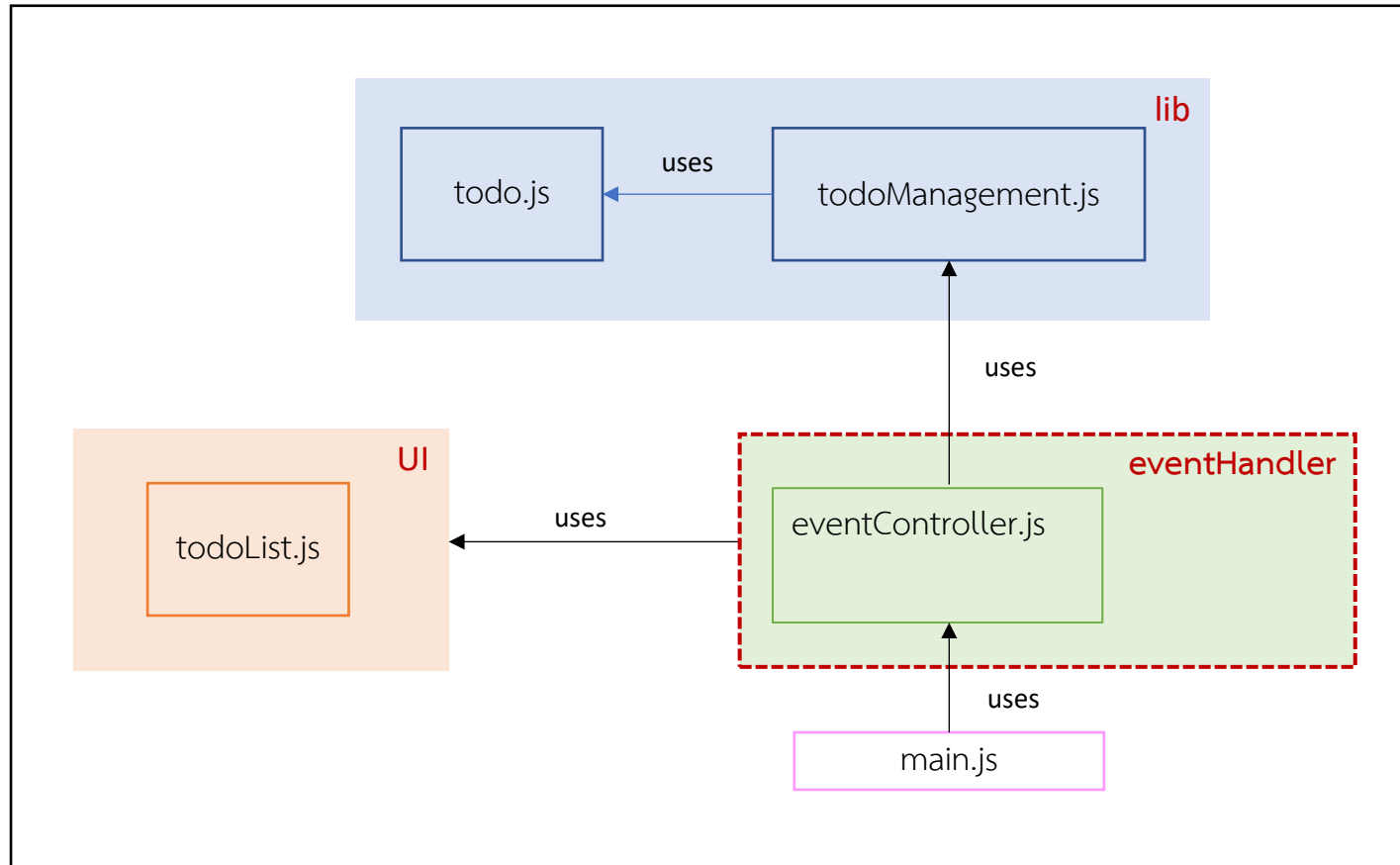




TodoList Application

Client-Side Storages

(Client Storages) TodoList Project Diagram



1. ไฟล์ `./lib/todo.js` เพื่อจัดการกับ property และ ฟังก์ชันของแต่ละ todo
2. ไฟล์ `./lib/todoManagement.js` เพื่อจัดการ โครงสร้างข้อมูลและฟังก์ชันของ todo ทั้งหมด
3. ไฟล์ `./UI/todoListUI.js` เพื่อจัดการ DOM ของ เอกสาร HTML
4. ไฟล์ `./eventHandler/eventController.js` เพื่อ จัดการกับ event ที่เกิดขึ้นบน DOM ของเอกสาร HTML และบันทึกข้อมูลของผู้ใช้ไว้ใน client-side storage
5. ไฟล์ `./main.js` เริ่มต้นทำงาน

ปรับปรุงไฟล์ `todoManagement.js` เพื่อจัดการโครงสร้างข้อมูลและฟังก์ชันของ `todo` ทั้งหมด

1. ให้เพิ่มฟังก์ชัน `loadTodos (userTodos)` เพื่อนำค่า `todo` ที่เก็บใน `local storage` เป็นค่าตั้งต้นให้กับ `todos array`

ปรับปรุงไฟล์ `eventController.js` เพื่อสร้าง `event handle functions` ดังนี้

1. ให้เพิ่มฟังก์ชัน `loadHandler()` เพื่อทำการ `get` ค่า `todos` จาก `local storage` และให้แปลง `string` เป็น `array` โดยใช้ `JSON.parse` โดยตรวจสอบว่าไม่เป็น `empty array` หรือ `null` หรือ `undefined` จากนั้นส่งไปเป็นพารามิเตอร์ของฟังก์ชัน `loadTodos()` จากนั้นให้เรียกฟังก์ชัน `getTodos()` เพื่อจะ `show` แต่ละรายการ `todos` ในเอกสาร `HTML` โดยการเรียกฟังก์ชัน `showTodoItem()` และให้ `add handler` ของปุ่ม `Not Done` และปุ่ม `remove` กรณี `todo` รายการนั้น `done` แล้ว ให้ใส่ `style background` เป็นสีเขียว และตัวอักษรเป็นสีขาว จากนั้น `update` จำนวนรายการ `done` และ `not done` โดยเรียกฟังก์ชัน `showNumberOfDone()` และ `showNumberOfNotDone()` ตามลำดับ และให้ทำการค้นหาปุ่ม `Add Todo` และลงทะเบียน `addTodoHandler()` ให้กับปุ่ม `Add Todo`
2. ให้เพิ่มฟังก์ชัน `beforeUnloadHandler(event)` โดยให้เรียกฟังก์ชัน `preventDefault()` เพื่อจะได้สามารถบันทึกค่า `todos` ปัจจุบัน โดยตั้งชื่อ `todos` เพิ่มเข้าไปใน `local storage` ได้ก่อนจะมีการ `unload HTML Page` และทำการเคลียร์ค่า `todo array` ให้เป็น `empty`

แทนที่ `code` ไฟล์ `main.js` ด้วยคำสั่ง ดังนี้

1. เพิ่ม `event` เมื่อ `window` ถูกโหลด โดยเรียกใช้ `loadHandler()` เพื่อโหลดค่า `todos` ของ `user` ที่บันทึกไว้ใน `local storage`
2. เพิ่ม `event` ก่อน `window` จะ `unload` โดยเรียกใช้ `beforeUnloadHandler(event)` เพื่อบันทึกค่า `todos` ของ `user` ใน `local storage`

Add Your Todo:

When window loading

Your Todo list

Grading Exam

Done

remove

Online Learning

Done

remove

Internship Meeting

Not Done

remove

Project Consulting

Done

remove

Number of Done:3

Number of Not Done:1

Application

Manifest

Service Workers

Storage

Storage

Local Storage

http://127.0.0.1:5501

Session Storage

IndexedDB

Web SQL

Cookies

Filter

Key	Value
todos	[[{"id":1,"description":"Grading Exam","done":true}, {"id":2,"description":"Online Learning","don...

When window beforeUnloading

▼ [{"id": 1, "description": "Grading Exam", "done": true}, {"id": 2, "description": "Online Learning", "done": true}, ...]

▶ 0: {id: 1, description: "Grading Exam", done: true}

▶ 1: {id: 2, description: "Online Learning", done: true}

▶ 2: {id: 3, description: "Internship Meeting", done: false}

▶ 3: {id: 4, description: "Project Consulting", done: true}