T.A. Marryshow Community College.

Computer and Electronics Department

A logo of a college

Description automatically generated

Project Report

|  |  |  |
| --- | --- | --- |
| Student Name | Ulric Aird | Kieran Francis |
| Student ID No. | AL2023014347 | Jd654321 |
| Course and No. | Data Structures and Algorithms (CIT-234) | |
| Lecturer | Mr. Christopher Mignon | |
| Project Title | Expensive Item Tracker | |
| Date | 8th November, 2024 | |

Contents

[1. Introduction 1](#_Toc181934867)

[1.1. Background 1](#_Toc181934868)

[1.2. Problem 1](#_Toc181934869)

[1.3. Objective and Scope 1](#_Toc181934870)

[2. Requirement analysis 2](#_Toc181934871)

[2.1. Feasibility study 2](#_Toc181934872)

[2.2. Budget estimate 2](#_Toc181934873)

[3. Design 3](#_Toc181934874)

[3.1. Project plan 3](#_Toc181934875)

[3.2. High level designs 3](#_Toc181934876)

[3.3. Algorithm 4](#_Toc181934877)

[3.4. Low level design 5](#_Toc181934878)

[4. Implementation 5](#_Toc181934879)

[5. Conclusion 6](#_Toc181934880)

[6. References 6](#_Toc181934881)

[7. Appendix 6](#_Toc181934882)

# Introduction

Our project aims to develop a software application that can determine if an item added to a list is considered expensive. This tool will be particularly useful for users who want to manage their budget efficiently by categorizing items based on their cost.

## Background

The idea behind this project stems from the need for better financial management tools. Many budgeting apps exist, but few offer the capability to automatically categorize items as expensive or not. Applications like Mint and YNAB provide financial tracking and budgeting features, but this project focuses specifically on the quick categorization of expenses to help users make informed purchasing decisions.

## Problem

The primary problem our project addresses is the difficulty users face in quickly identifying and categorizing their expenses. Without a quick way to identify expensive items, users may struggle to manage their budgets effectively, leading to overspending and financial stress.

## Objective and Scope

**Objectives:**

* Develop a user-friendly interface for adding items to a list.
* Implement an algorithm to determine if an item is expensive based on user-defined criteria or historical data.
* Provide real-time feedback to the user about the categorization.

**Milestones:**

* Complete the design of the user interface.
* Implement and test the categorization algorithm.
* Conduct user testing and gather feedback.
* Finalize and launch the software.

# Requirement analysis

To develop this software, several resources and technologies are needed:

* **Programming Languages:** JavaScript
* **Frameworks:** React frontend
* **Database:** Browser Local Storage for Database
* **Platforms:** Web-based, deployable on any web server

## Feasibility study

The feasibility study involves evaluating the technical, economic, and operational feasibility of the project:

* **Technical Feasibility:** The technologies and frameworks selected are well-documented and widely used, ensuring ample resources and community support.
* **Economic Feasibility:** The cost of development is minimized by using open-source technologies. The primary investment is developer time.
* **Operational Feasibility:** The project aims to meet a common user need, increasing its likelihood of acceptance and usage.

## Budget estimate

The budget for this project primarily involves the cost of development and deployment:

* **Development Costs:** Estimated at $50 - $100, considering developer salaries and time.
* **Hosting and Maintenance:** Around $10 - $20 per year, depending on the hosting provider and scale of usage.
* **Miscellaneous Costs:** $30 for additional resources, tools, or unforeseen expenses.

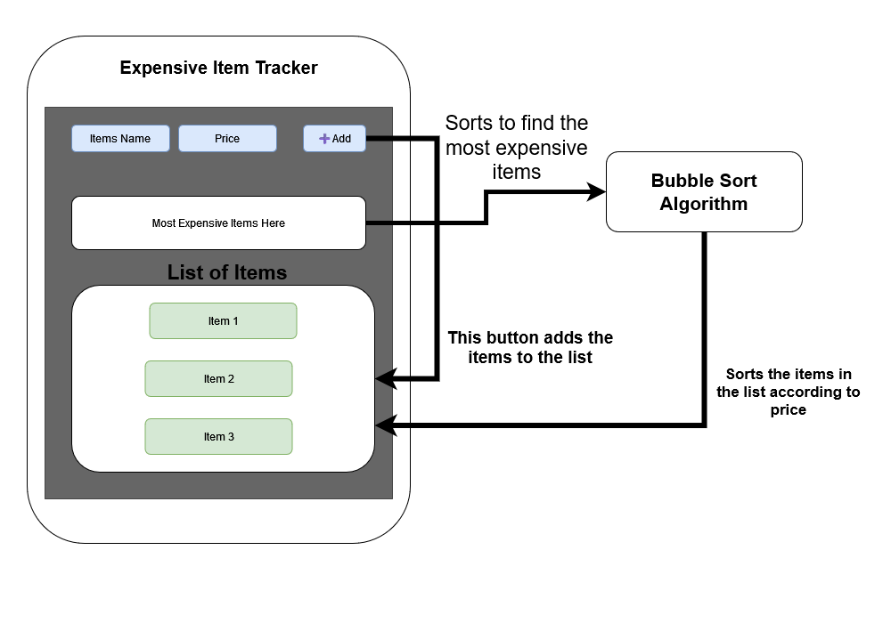
This detailed breakdown should provide a clear understanding of the project’s goals, requirements, and feasibility. Any other details you need help with?

# Design

## Project plan

Project is expected to be completed within a week or 2 weeks for the most.

## High level designs

This is a high-level interactive design of how the Expensive item tracker supposed to look and operate.

## Algorithm

For this algorithm, we will be using the Array and Dictionary (Object) Data Structure and Bubble Sort Algorithm.

Module BubbleSort(Array arr) {

For (int i = 0; i < arr.length; i++) {

For (int j = I+1; j < arr.length; j++) {

If (arr[j].price > arr[i].price) {

Array Temp\_data = arr[i];

arr[i] = arr[j];

arr[j] = Temp\_data;

}

}

}

}

Module Display\_Expensive(Array arr) {

Output arr[0].name, “ is the most expensive item”;

}

Array items = [

{ id: 1, name: “Watch”, price: 90.2 },

{id: 2, name: “Sweater”, price: 80 },

];

BubbleSort(items);

Display\_Expensive(items);

## Low level design

A diagram of a flowchart

Description automatically generatedThis is a flow chart of how the algorithm should be implemented within the project.

# Implementation

# Conclusion

The Expensive Item Tracker project developed to aid users to keep track of their finance by categorizing items as expensive according to certain predefined criteria. The tool, constructed in JavaScript, React, and browser local storage, enables real-time expense insights not available through current budgeting applications. Its low-cost development is based on open-sourced tool development and a bubble sort algorithm that identifies items as the 'priciest'. This online solution will be developed in a user-friendly way to make the solution fast and useful, allowing users to make confident expenditure decisions and enhance budgetary management further.

# References

**There are no sources in the current document.**

# Appendix