



THOMPSON RIVERS UNIVERSITY

Algorithms & Data Structure (Fall 2022) - SENG 3110

URepair Household Equipment Repair Program

Bradley D. Schmidt (T00711584)

Amirmohammad Ghourchian (T00703368)

Salamudin Abdullahi (T00697855)

Project Title

SENG 3110

Date submitted

Table of Contents

1	Introduction.....	5
2	Design Problem.....	6
2.1	Problem Definition.....	6
2.2	Design Requirements.....	6
2.2.1	Functions.....	6
2.2.2	Objectives.....	6
2.2.3	Constraints.....	6
3	Solution.....	7
3.1	Solution 1.....	7
3.2	Solution 2.....	7
3.3	Final Solution.....	7
3.3.1	Features.....	7
3.3.2	Environmental, Societal, Safety, and Economic Considerations.....	8
3.3.3	Limitations.....	8
4	Team Work.....	8
4.1	Meeting 1.....	8
4.2	Meeting 2.....	8
4.3	Meeting 3.....	8
4.4	Meeting 4.....	8
5	Project Management.....	9
6	Conclusion and Future Work.....	10
7	References.....	11
8	Appendix.....	12

- The table of contents should be automatically generated by selecting "References/ Table of Contents." Remember that the table of contents should not have an entry of the "Table of Contents" itself.
- Proofread the text for typing and grammar mistakes.
- Follow the IEEE Bibliography style for the references by selecting "References/ Citations & Bibliography/ Style".

List of Figures

List of Tables

1 Introduction

1.1 Scope and Background

Currently, getting household equipment repaired can be frustrating due to the lack of reliable and affordable repair options. URepair, a startup in Canada, is working to solve this issue by creating a platform that connects people with repair professionals. In this project, we will design and build a **Household Equipment Repair Program** using C/C++ and various algorithms. The program will allow users to submit repair requests, get responses from repair talents, and track the performance of the repair professionals. The goal is to make repairs a more accessible, and interactive database for everyone involved. This is designed to be a proof of concept, rather than a complete business.

1.2 Goal and Method

The goal of this project is to create a program/platform that connects users with repair talents. The idea is to make it easier for customers to quickly find and book repair services for household appliances. This way, it solves the problem of having to search for reliable help when something breaks. We'll be using some advanced techniques, like algorithms and data structures, to make sure the system matches customers with technicians efficiently, so they can get fast and high-quality service.

1.3 Report Objectives

In this report, we'll go over how we designed and developed the platform, focusing on the problem it addresses, the design requirements, and how we plan to achieve our goals while working within the project's limitations/constraints.

1.4 Role of Algorithms and Data Structures

Through the use of the new algorithms and data structures we have learned, we will create a more efficient solution than previously possible. By implementing data structures properly we will ensure that our algorithms will work properly and fast. By implementing proper algorithms for searching, we will hopefully create a perfectly seamless experience for the end user.

1.5 Stakeholders and Benefits

The end user will be gaining a more reliable service than previously available, with completely transparent quality and pricing due to our sorting algorithms. Repair professionals will have increased visibility and job opportunities especially when starting out because of the different sorting methods for people looking to get a job done. Contractors will also be able to build and showcase their reputation via their profile and badges, etc.

2 Design Problem

2.1 Problem Definition

The main problem is that it's hard for homeowners (customers) to find trustworthy and affordable repair professionals for their household equipment. Many people don't know where to look, and when they do find someone, they aren't sure about the quality of the service. URepair wants to make this process simpler by building a platform where people can communicate with repair talents, submit repair requests and get bids from verified repair talents. The platform will also provide a way to see the performance of each repair professional based on previous jobs, making it easier for customers to make good decisions.

2.2 Design Requirements

2.2.1 Functions

The program will perform the following functions:

- **Submit Repair Requests:** Users/customers will be able to submit requests for repairs.
- **Track Talent Responses:** The program will keep track of repair talents who respond to customer requests.
- **Manage Repair History:** Store information on repair jobs completed by each talent and make it available to users.
- **Handle User Data:** Customers can add, update/edit, or delete their contact information in the system.
- **Retrieve Data:** The system will be able to pull information such as the best-rated repair talents and active repair requests.

2.2.2 Objectives

The design will focus on:

- **Ease of Use:** The program will have a user-friendly interface that makes it simple for anyone to use (even with little digital knowledge).
- **Reliability:** It will ensure that users can count on accurate information when selecting a repair professional.
- **Scalability:** it will be built in a way that allows it to grow as more users join (adaptable).
- **Security:** Users' personal information will be protected and only accessible to authorized personnel.
- **Fast Performance:** The system will provide quick access to the data it holds, allowing users to easily view and manage their requests.

2.2.3 Constraints

The project will be bound by the following constraints:

- **Use of C/C++:** The program must be written in C or C++.
- **Data Structures:** We must use at least three different data structures, like arrays, linked lists, or binary search trees, to store and manage information.
- **Compliance:** The system must follow Canadian privacy laws to protect user data.
- **Performance:** The system needs to process and return data quickly enough to ensure a good user experience.
- **Limited Resources:** The program must work within the available memory and processing power to avoid slowdowns.

3 Solution

This section will provide an account of some solutions your team brainstormed to implement the project. Some solutions might not have all the desired features, and some might not satisfy the constraints or both. These solutions come up in your mind while you brainstorm ways of implementing all the features while meeting the constraints. Towards, the end you select a solution that you think has all the features and satisfies all the constraints. Remember that an engineering design is iterative in nature!

3.1 Solution 1

Write a brief description of your first solution and provide the reasons for not selecting this one.

3.2 Solution 2

This is an improved solution but might not be the final solution that you select. Give a brief description of this solution here.

3.3 Solution 3

This is an improved solution but might not be the final solution that you select. Give a brief description of this solution here.

3.4 Final Solution

This is the final solution. Explain why it is better than other solutions. You may use a table for comparison purposes. After providing the reason for selecting this solution, detail it below.

This is the final solution. Explain why it is better than other solutions. You may use a decision matrix chart for comparison purposes.

Table I Decision matrix chart for the considered alternatives

		Solutions					
		Solution 1		Solution 2		Final Solution	
Criteria	Weight	Score	Partial Score	Score	Partial Score	Score	Partial Score
Criterion 1	0.40	6/10	0.240	6/10	0.240	7/10	0.280
Criterion 2	0.25	3/5	0.150	4/5	0.200	5/5	0.250
Criterion 3	0.20	9/15	0.120	12/15	0.160	9/15	0.120
Criterion 4	0.15	9/10	0.135	6/10	0.090	8/10	0.120
Sum	1.00		0.645		0.690		0.77

3.4.1 Features

Give an account of all the features your prototype has and which functions or methods will enable those features. These features may be tabulated (with a title) for improved comprehension.

3.4.2 Environmental, Societal, Safety, and Economic Considerations

Explain how your engineering design took into account environmental, societal and economic considerations. It may include how your design has positive contributions to the environment and society. What type of economic decisions you made. How did you make sure that the design is safe to use etc.

3.4.3 Limitations

Every product has some limitations, and so is the case with your design product. Highlight some of the limitations of your prototype here.

4 Team Work

Since this is a group project, you must have a fair distribution of tasks among yourselves. To this end, you must hold meetings to discuss the distribution of tasks and keep track of the project progress.

4.1 Meeting 1

Time: September 22, 2022, 10:00 am to 11:00 am

Agenda: Distribution of Project Tasks

Team Member	Previous Task	Completion State	Next Task
Team member 1	N/A	N/A	Task 1
Team member 2	N/A	N/A	Task 2
Team member 3	N/A	N/A	Task 3

4.2 Meeting 2

Time: September 29, 2022, 10:00 am to 11:00 am

Agenda: Review of Individual Progress

Team Member	Previous Task	Completion State	Next Task
Team member 1	Task 1	80%	Task 1, Task 5
Team member 2	Task 2	50%	Task 2
Team member 3	Task 3	100%	Task 6

4.3 Meeting 3

Provide a similar description here.

4.4 Meeting 4

Provide a similar description here.

5 Project Management

Provide a Gantt chart showing the progress of your work here. Mention all the tasks along with their predecessors. Provide the slack time of each task and identify the critical path.

6 Conclusion and Future Work

- A summary of what you achieved. Mention all the design functions and objectives that you achieved while satisfying the constraints?
- While keeping the limitations of your prototype, provide recommendations for future design improvements.

7 References

- Use the IEEE reference style.
- Do not put any reference if it is not cited in the text.

8 Appendix

If you want to provide an additional information, use this appendix.