

Projeto LCOM 2019/20

The Sealed Room

Breno Accioly de Barros Pimentel
201800170

Índice

1.....	Instruções
1.1.....	Menu Inicial
1.2.....	Jogo
1.3.....	LeaderBoard
1.4.....	Menu de Ajuda
2.....	Status do Projeto
3.....	Estrutura
3.1.....	Módulos
3.2.....	Call Graph
4.....	Detalhes de Implementação
5.....	Conclusão

1.Instruções

1.1 Menu Inicial

A aplicação começa por apresentar o menu inicial. Nele é possível explorar as diferentes funcionalidades do programa.

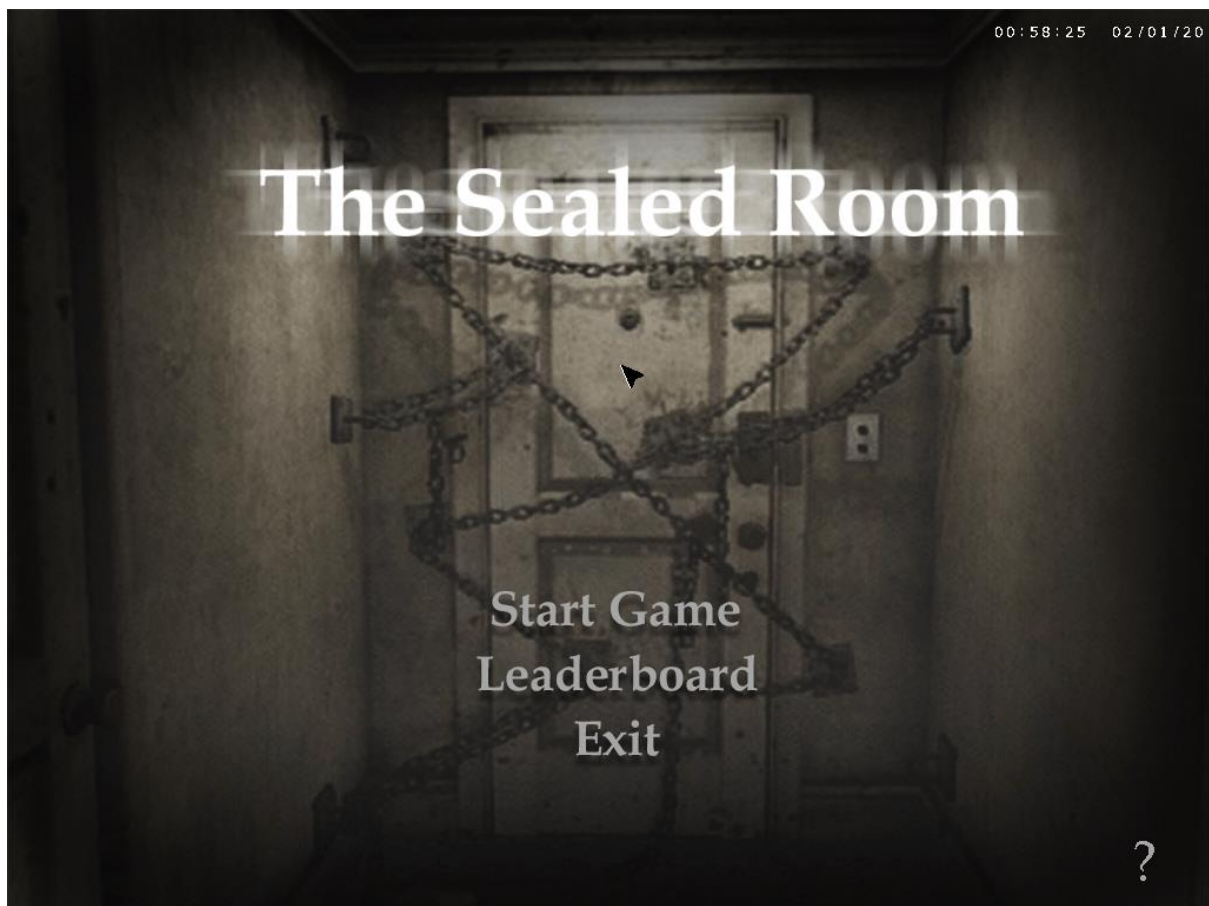


Figura 1 – Tela inicial do jogo.

Start Game: Inicia o jogo

LeaderBoard: Apresenta o placar de líderes

Exit: Fecha o jogo (também é possível sair com ESC)

Help (Canto inferior direito): Apresenta instruções e informações adicionais

Relógio(Canto superior direito): Hora e data do sistema

1.2 Jogo



Figura 2 – Aplicação sendo utilizada por um jogador.

O objetivo do jogo está em achar 7 chaves para poder escapar do labirinto. Existe um tempo limite para isso e o jogador deve evitar armadilhas.

A movimentação é realizada utilizando as teclas W,A,S,D. Informações de tempo e chaves coletadas estão no canto superior direito. A tecla ESC termina o jogo e retorna para o menu inicial.

Após ter coletado as 7 chaves é apresentado ao jogador a tela de resultados e, posteriormente, um menu onde poderá escrever seu nome para ir ao placar de líderes.



Figura 3 – Tela final do jogo com as chaves adquiridas e o tempo restante.

1.3 LeaderBoard

Tabela com os melhores jogadores. O ranking considera o tempo necessário para concluir o desafio como fator de ordenação.

Para ir ao placar de líderes é necessário coletar todas as chaves. O usuário terá que escrever seu nome usando 5 letras e pressionar ENTER, é possível usar BACKSPACE para eliminar erros. Caso não deseje salvar seu nome, pode ser pressionado a tecla ESC que a aplicação voltará ao menu inicial sem nenhuma alteração ao placar.

Após guardar o nome e tempo, se a pontuação for melhor do que uma das outras 5 do placar, essa poderá agora ser visualizada a partir da opção Leaderboard no menu inicial.

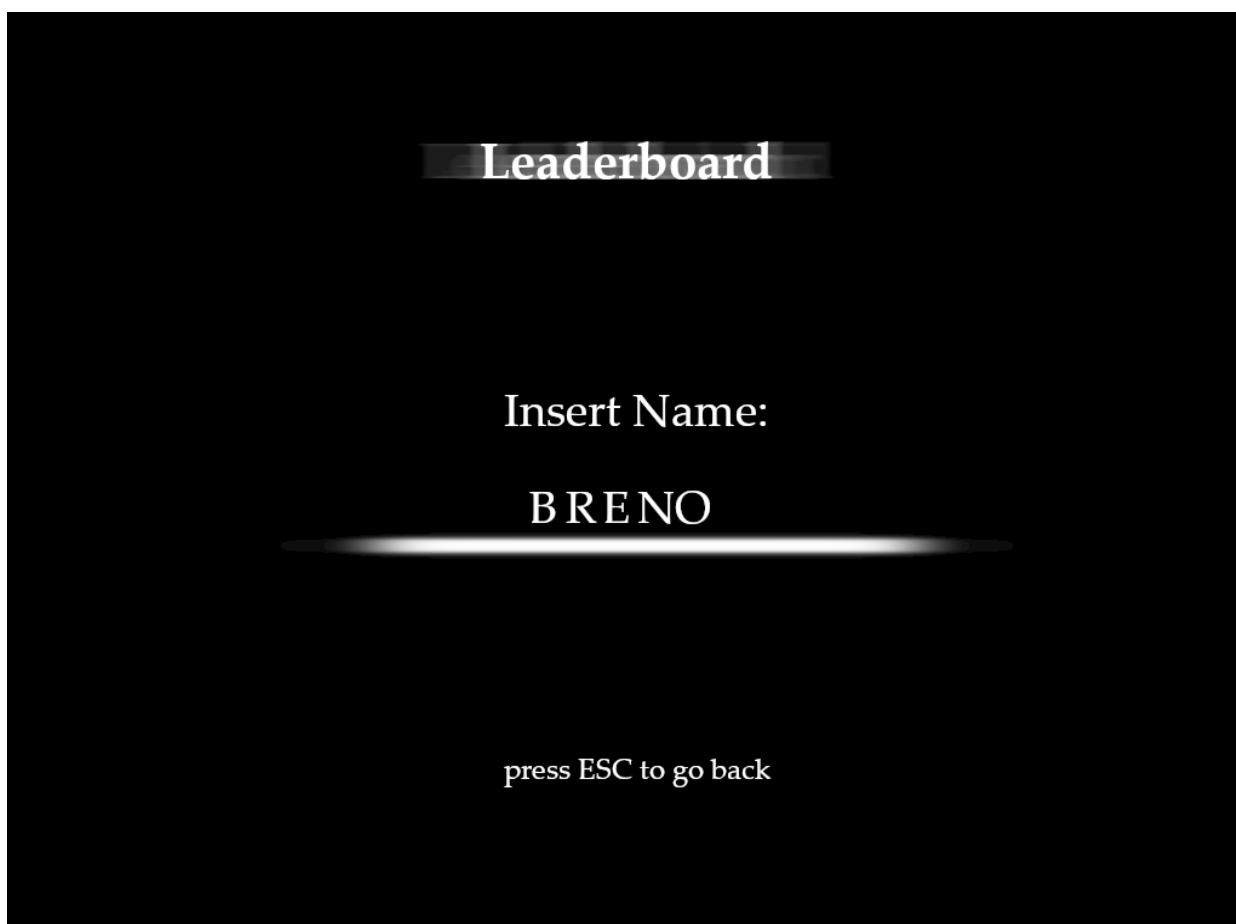


Figura 4 – Leaderboard com o nome do jogador.

Leaderboard	
	time
1. B R E N O	7 0
2. O S C A R	8 0
3. S I M B A	8 1
4. L C O M C	8 3
5. A S D F F	8 5
press ESC to go back	

Figura 5 – Leaderboard com cinco jogadores com as melhores colocações.

1.4 Menu de Ajuda

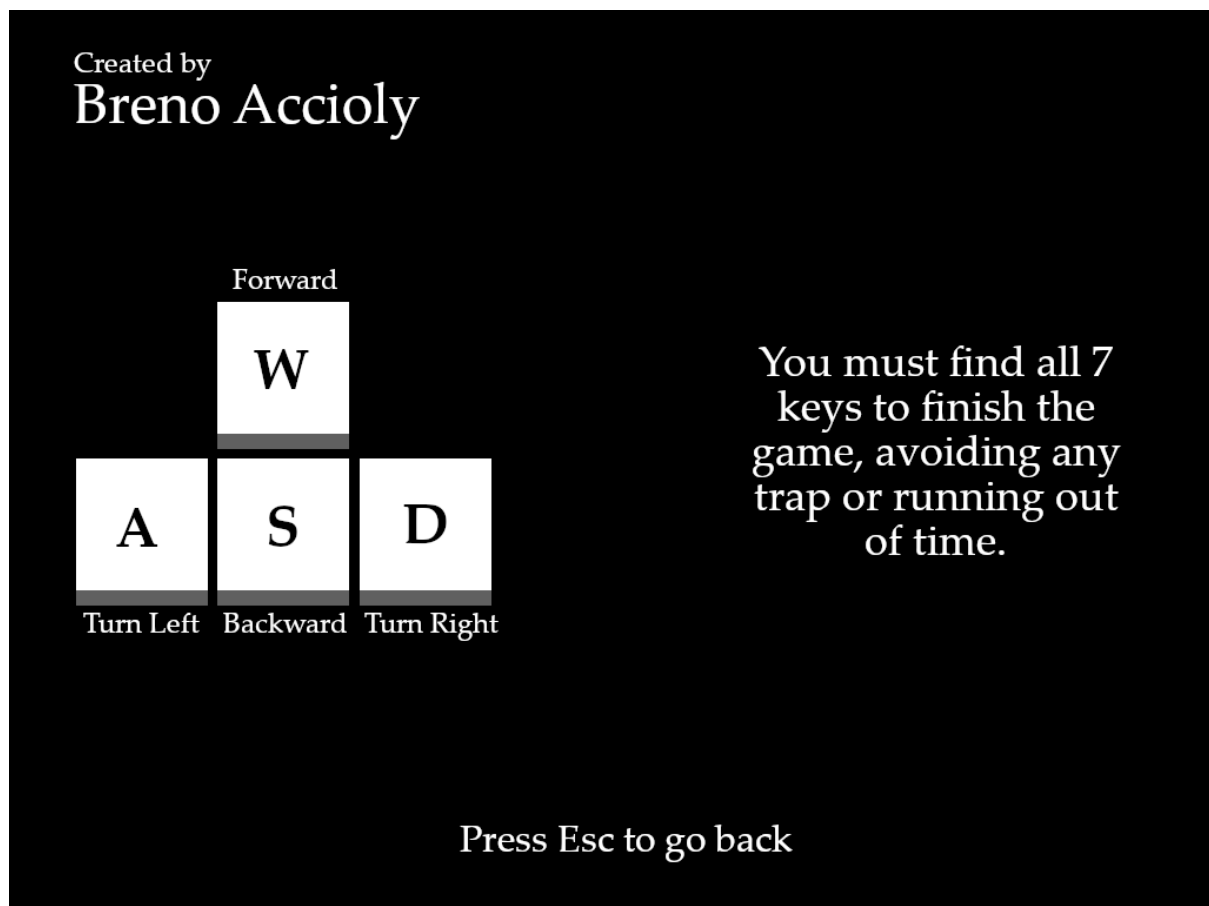


Figura 6 – Tela do Menu de Ajuda informações trívias para a aplicação.

O menu de ajuda apresenta informações a respeito do jogo, como a movimentação e o objetivo do jogo. É possível ainda testar as teclas e verificar a sua animação no ecrã.

2. Status do Projeto

Serial Port - Seria utilizado para o modo multijogador, entretanto não foi implementado no projeto.

Dispositivos Usados

Dispositivo	Usado para	Int.
Timer	Controlar o frame rate e para marcar o tempo do jogo	S
KBD	Movimentação do jogo, retroceder e leaderboard	S
Mouse	Selecionar opções no menu inicial	S
Video Card	Gráficos dos menus e sprites do jogo	S
RTC	Relógio e data no menu inicial	N

Tabela 1 – Tabela contendo os dispositivos usados.

Timer:

Fundamental para o desenvolvimento do jogo, o Timer foi responsável por atualizar os gráficos, como a movimentação e animação do personagem e armadilhas em **game.c**, como também controlar a quantidade de quadros por segundo e dinâmicas do jogo.

KBD :

Foi usado em todos os menus, com a tecla ESC é possível retroceder ou sair do jogo a qualquer momento. A movimentação do personagem foi feita a partir das teclas W, A, S, D e no placar de líderes o teclado foi usado para inserção do nome do jogador.

No código é usado tanto interrupções como também *polling* dependendo do contexto. Em **help_menu(mainMenu.c)** e **startMenu(mainMenu.c)** é possível observar as diferentes técnicas utilizadas.

Mouse :

O mouse foi usado no menu principal (**mainMenu.c**) para seleção dos diferentes botões apresentados. Foi também feita uma estrutura que armazena informações de cada botão, como sua posição e tamanho, sendo assim mais fácil de verificar os limites para que quando o cursor estiver sob este, ele seja ativado.

Video Card :

Usado na apresentação gráfica de todos os menus e jogo. Para isso foi usado o modo de vídeo 0x118, com resolução 1024x768 e cores 8:8:8.

Foram feitas leituras de XPM e foi utilizado também a técnica de double buffering, com o objetivo de deixar o jogo mais fluído.

O sistema de colisões, como em **vg_verify_collision(vídeo.c)**, foi possível pela detecção de uma linha de determinada cor que limitava o espaço onde seria possível andar pelo jogador e também para verificar se o jogador foi atingido por uma armadilha. Foi também utilizado uma cor especial para verificar transparência.

RTC :

Realiza a leitura de data e hora do sistema e apresenta essas informações no menu principal. Utilizado principalmente na função **draw_time(mainMenu.c)**.

3. Estrutura

3.1 Módulos

Todos os módulos implementados por Breno Accioly.

MainMenu (20%)

Módulo responsável por inicializar e apresentar as funcionalidades da aplicação, entres elas, Jogar, Placar de Líderes, Menu de Ajuda e a apresentação da data e hora.

Possui ainda a estrutura **button**, que armazena uma imagem e seu sprite, além do posicionamento horizontal e vertical e suas dimensões.

Game (20%)

Nesse módulo é onde o jogo realmente acontece. Possui diferentes funções para toda a dinâmica do jogo.

Nele, existe a estrutura **player** que guarda o nome e tempo de conclusão do desafio quando o jogador termina. Auxiliando na sua inserção ao placar de líderes.

Maze (15%)

Gera um labirinto aleatoriamente para ser usado no jogo e possui funções complementares para fazer sua apresentação no ecrã e adicionar chaves e armadilhas também de forma aleatória.

RTC (4%)

Faz a leitura da data e hora do sistema que serão apresentadas no menu inicial.

KBD (10%)

O dispositivo foi usado principalmente para o controle do personagem e para inserir o nome no placar de líderes. Este foi importado do laboratório 3.

Mouse (10%)

Utilizado no menu inicial, com esse módulo, é feita a interação do usuário com botões. Esse módulo foi importado do laboratório 4.

Timer (5%)

Foi importado do laboratório 2 e é responsável pelo controle de frames e atualização de estados do jogo como, por exemplo, para verificar se uma armadilha está ativada ou não.

Proj (2%)

Responsável por inicializar o modo gráfico (vg_init) e iniciar o menu principal (startMenu), dando início assim na aplicação.

I8042 (2%)

Esse módulo foi importado do laboratório 5, nele foram adicionadas informações como *make codes*.

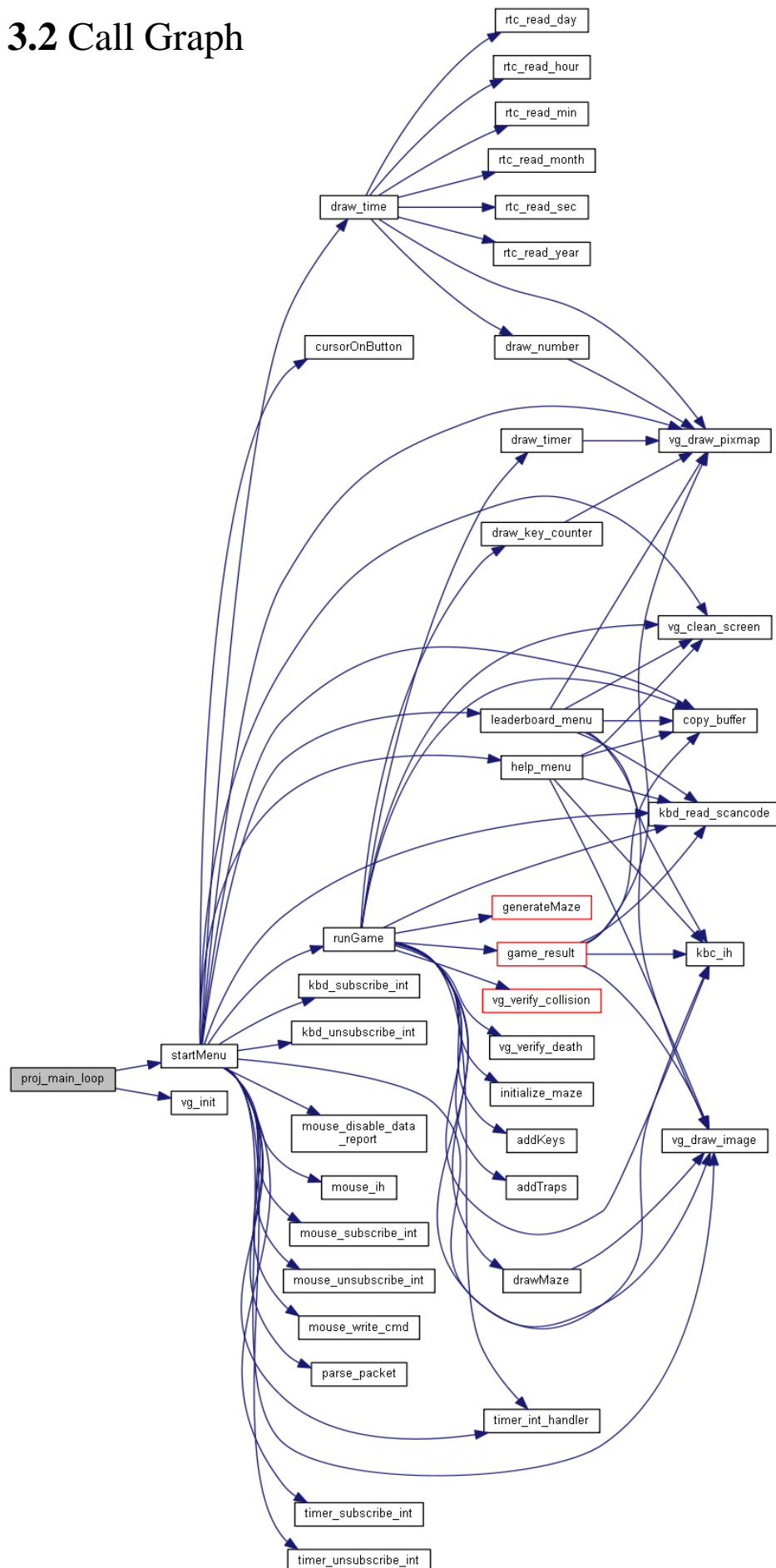
i8254 (2%)

Esse módulo foi importado do laboratório 2 e não sofreu grandes alterações.

Video (10%)

Foi importado do laboratório 5 mas possui modificações importantes para o desenvolvimento da aplicação, como a verificação de colisão e o *double buffering*.

3.2 Call Graph



4. Detalhes de Implementação

O labirinto de dimensões 1680 pixels por 1520 pixels foi possível através de um algoritmo de busca em profundidade, com alterações para ser possível posicionar aleatoriamente as armadilhas e chaves.

A fim de criar uma maior dificuldade, foi também decidido que o jogador não poderia visualizar todo mapa do jogo. Para resolver esse problema, a movimentação teve que ser realizada tendo o labirinto como base, onde este se move e não o personagem, que fica parado no centro da tela apenas mudando as direções do seu sprite gerando a impressão de movimento. A função `drawMaze` precisou então de argumentos com o deslocamento horizontal e vertical, atualizados a cada movimentação do jogador e responsáveis por posicionar o labirinto na posição correta do ecrã.

O sistema de colisões teve que ser pensado para um cenário gerado aleatoriamente. Para isso, foi necessário incluir uma linha de cor diferenciada em cada xpm que representasse as paredes e limites do jogo, como também as regiões de armadilha onde o jogador não poderia passar sem ativar o “Fim de Jogo”. A cada movimentação do personagem é feita a verificação se o mesmo se encontra posicionado nessas cores e retorna a informação se o jogador encontrou uma barreira ou foi atingido por uma armadilha.

Foi também realizada a técnica de Double Buffer, com o objetivo de diminuir anomalias no ecrã e deixar o jogo mais fluído.

5. Conclusão

O projeto foi animador porquê foi dada uma certa liberdade de desenvolvimento e também por ser uma boa oportunidade de aplicar o conhecimento adquirido durante as aulas. E, além do desafio do projeto em si, também tive de o fazer sozinho o que dificultou em certas partes, entretanto, fui capaz de finalizar este e me sinto contente com o resultado mesmo reconhecendo que certas partes poderiam estar melhores e que infelizmente não tive tempo de mudá-las.

Como sugestão, indicaria uma melhor organização nos conteúdos, uma vez que tive dificuldade em buscar informações necessárias nesses.

Por fim, o trabalho representou um grande desafio e foram dedicadas muitas horas semanais para sua realização, e a cada hora trabalhada aprendi imenso.