

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO  
INSTITUTO DE COMPUTAÇÃO  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

JOYCE DE SANT' ANNA BRUM  
THIAGO OUTEIRO PEREIRA DAMASCENO

ELABORAÇÃO E IMPLEMENTAÇÃO DE JOGO EDUCATIVO PARA ENSINO DE  
SQL

RIO DE JANEIRO  
2022

JOYCE DE SANT' ANNA BRUM  
THIAGO OUTEIRO PEREIRA DAMASCENO

ELABORAÇÃO E IMPLEMENTAÇÃO DE JOGO EDUCATIVO PARA ENSINO DE  
SQL

Trabalho de conclusão de curso de graduação  
apresentado ao Instituto de Computação da  
Universidade Federal do Rio de Janeiro como  
parte dos requisitos para obtenção do grau de  
Bacharel em Ciência da Computação.

Orientador: Prof. Geraldo Bonorino Xexeo  
Co-orientador: Prof. Luiz Felipe Costa

RIO DE JANEIRO

2022

B893e

Brum, Joyce de Sant' Anna

Elaboração e implementação de jogo educativo para ensino de SQL /  
Joyce de Sant' Anna e Thiago Outeiro Pereira Damasceno. – 2022.

155 f.

Orientador: Geraldo Bonorino Xexeo.

Coorientador: Luiz Felipe Costa.

Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação)  
- Universidade Federal do Rio de Janeiro, Instituto de Computação, Bacharel  
em Ciência da Computação, 2022.

1. SQL. 2. Game. 3. Ensino de SQL. 4. Jogo investigativo. 5. Educação em  
TI. 6. Jogo educativo. I. Damasceno, Thiago Outeiro Pereira. II. Xexeo,  
Geraldo Bonorino (Orient). III. Costa, Luiz Felipe (Coorient.). IV.  
Universidade Federal do Rio de Janeiro, Instituto de Computação. V. Título.

JOYCE DE SANT' ANNA BRUM  
THIAGO OUTEIRO PEREIRA DAMASCENO

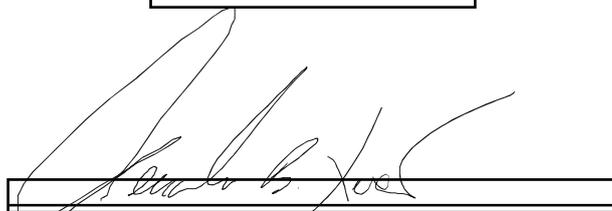
ELABORAÇÃO E IMPLEMENTAÇÃO DE JOGO EDUCATIVO PARA ENSINO DE  
SQL

Trabalho de conclusão de curso de graduação  
apresentado ao Instituto de Computação da  
Universidade Federal do Rio de Janeiro como  
parte dos requisitos para obtenção do grau de  
Bacharel em Ciência da Computação.

Aprovado em 30 de Setembro de 2022

BANCA EXAMINADORA:

[Redacted]



Geraldo Bonorino Xexéo  
DSc (COPPE/UFRJ)

Participação por videoconferência

Giseli Rabello Lopes  
PhD (IC/UFRJ)

Participação por videoconferência

Eduardo Freitas Mangeli de Brito  
MSc (IC/UFRJ)

Dedicatória: Dedicamos esse trabalho aos nossos pais, que nos apoiaram em toda a nossa trajetória acadêmica e aos nossos professores, que nos orientaram e nos ensinaram tudo o que precisávamos em nosso período na universidade.

## AGRADECIMENTOS

Agradecemos ao nosso professor orientador Geraldo Xexeo por toda a orientação e ajuda dada no desenvolvimento desse trabalho, ao nosso antigo parceiro de projeto, Gabriel Matos, que acabou saindo no meio mas que contribuiu enormemente para a elaboração da história e da proposta e ao professor Luiz Felipe Costa que, apesar de não ter nos acompanhado no projeto até o final, nos ajudou na concepção geral do jogo.

*"This is the real secret of life – to be completely engaged with what you are doing in the here and now. And instead of calling it work, realize it is play."*

**Alan Watts**

## RESUMO

Este trabalho apresenta um jogo educacional — denominado SQLGame — baseado no conceito de narrativa interativa. Neste jogo, o jogador tem como objetivo resolver crimes tendo como fonte de pesquisa uma base de dados e ferramenta de busca os comandos SQL (Sequel). O jogo abrange todo o conteúdo didático DQL (Data Query Language), tendo como objetivo que o jogador, ao terminá-lo, saiba executar com facilidade comandos de consulta SQL.

**Palavras-chave:** sql; game; Ensino de SQL; Jogo Investigativo; Educação em TI; jogo educativo; jogo serio; sqlGame

## ABSTRACT

This article presents an educational game — named as SQLGame — based on the concept of an interactive narrative. In this game, the player's objective is to solve crimes having, as search source, a data base and, as search tool, the SQL commands. The game covers all DQL didactic content, besides its objective is that the player knows how to do any SQL query command at the end of the game.

**Keywords:** sql; game; SQL teaching; Investigative Game; education; edutainment game; serious game; learn sql; sqlGame

## LISTA DE ILUSTRAÇÕES

Figura 1 – Modelo ENDO-GDC	23
Figura 2 – Cartas de Dimensão de Conhecimento	25
Figura 3 – Cartas de Processo Cognitivo	26
Figura 4 – Fluxo didático do Learning SQL	37
Figura 5 – Fluxo didático do SQL para Leigos	38
Figura 6 – Fluxo didático do Use a Cabeça! SQL	38
Figura 7 – Modelo ENDO-GDC Preenchido	48
Figura 8 – 6-11 <i>Framework</i>	49
Figura 9 – Tela Principal	55
Figura 10 – Tela de Contatos	55
Figura 11 – Tela de Mensagens	56
Figura 12 – Tela de Pistas	56
Figura 13 – Tela de Tabelas	57
Figura 14 – Fluxo Didático SQLGame	57
Figura 15 – Tela Inicial	60
Figura 16 – Tutorial	60
Figura 17 – Tutorial com Foco	61
Figura 18 – Botão em foco	61
Figura 19 – Pista em foco	61
Figura 20 – Tela Chat	62
Figura 21 – Tela Pistas	62
Figura 22 – Tela Pistas mais Complexa	63
Figura 23 – Tela de Banco de Dados	64
Figura 24 – SQL Game no itch.io	65
Figura 25 – Resposta Questionário: Autorização de Participação	82
Figura 26 – Resposta do Questionário: Faixa Etária	82
Figura 27 – Resposta do Questionário: Sexo	83
Figura 28 – Resposta do Questionário: Curso de Graduação	83
Figura 29 – Resposta do Questionário: Frequência de Jogos Digitais	83
Figura 30 – Resposta do Questionário: Frequência de Jogos não Digitais	83
Figura 31 – Resposta do Questionário: Design do Jogo Atraente	84
Figura 32 – Resposta do Questionário: As Cores Combinam	84
Figura 33 – Resposta do Questionário: Rápido de Aprender	84
Figura 34 – Resposta do Questionário: Fácil de Aprender	85
Figura 35 – Resposta do Questionário: Outros Aprenderiam Fácil	85
Figura 36 – Resposta do Questionário: Fácil de Jogar	85

Figura 37 – Resposta do Questionário: Regras Claras . . . . .	86
Figura 38 – Resposta do Questionário: Fontes Legíveis . . . . .	86
Figura 39 – Resposta do Questionário: Cores Compreensíveis . . . . .	86
Figura 40 – Resposta do Questionário: Confiante sobre Aprender . . . . .	87
Figura 41 – Resposta do Questionário: Adequadamente Desafiador . . . . .	87
Figura 42 – Resposta do Questionário: Novos Desafios . . . . .	87
Figura 43 – Resposta do Questionário: Não Monótono . . . . .	88
Figura 44 – Resposta do Questionário: Realização . . . . .	88
Figura 45 – Resposta do Questionário: Esforço Pessoal . . . . .	88
Figura 46 – Resposta do Questionário: Satisfação . . . . .	89
Figura 47 – Resposta do Questionário: Recomendação . . . . .	89
Figura 48 – Resposta do Questionário: Interação Social . . . . .	89
Figura 49 – Resposta do Questionário: Cooperação/Competição . . . . .	90
Figura 50 – Resposta do Questionário: Boa Interação . . . . .	90
Figura 51 – Resposta do Questionário: Diversão . . . . .	90
Figura 52 – Resposta do Questionário: Alegria . . . . .	91
Figura 53 – Resposta do Questionário: Interesse . . . . .	91
Figura 54 – Resposta do Questionário: Imersão - Noção do tempo . . . . .	91
Figura 55 – Resposta do Questionário: Imersão - Esquecer do ambiente . . . . .	92
Figura 56 – Resposta do Questionário: Relevância . . . . .	92
Figura 57 – Resposta do Questionário: Relacionado a SQL . . . . .	92
Figura 58 – Resposta do Questionário: Bom Método . . . . .	93
Figura 59 – Resposta do Questionário: Preferência . . . . .	93
Figura 60 – Resposta do Questionário: Contribuição na Aprendizagem . . . . .	93
Figura 61 – Resposta do Questionário: Eficiência na Aprendizagem . . . . .	94
Figura 62 – Resposta do Questionário: Objetivo de Aprendizado 1 . . . . .	94
Figura 63 – Resposta do Questionário: Objetivo de Aprendizado 2 . . . . .	94
Figura 64 – Resposta do Questionário: Objetivo de Aprendizado 3 . . . . .	95
Figura 65 – Resposta do Questionário: Objetivo de Aprendizado 4 . . . . .	95
Figura 66 – Resposta do Questionário: Objetivo de Aprendizado 5 . . . . .	95

## LISTA DE TABELAS

Tabela 1 – Bancos de dados mais populares do mundo e seus rankings . . . . .	36
--	----

## LISTA DE QUADROS

Quadro 1 – Diferenças entre jogos de entretenimento e jogos sérios . . . . .	19
Quadro 2 – Tipos de jogos sérios . . . . .	20
Quadro 3 – Definição das dimensões/subdimensões . . . . .	32
Quadro 4 – Comparação Jogos Educativos SQL . . . . .	40
Quadro 5 – Tendência Central das Dimensões Avaliadas . . . . .	66
Quadro 6 – Tendência Central das Perguntas . . . . .	96

## LISTA DE ABREVIATURAS E SIGLAS

DQL	Data Query Language - Linguagem de Consulta de Dados
DML	Data Manipulation Language - Linguagem de Manipulação de Dados
DDL	Data Definition Language - Linguagem de Definição de Dados
DCL	Data Control Language - Linguagem de Controle de Dados
DTL	Data Transaction Language - Linguagem de Transação de Dados
POC	Proof of Concept - Prova de Conceito - é uma forma de provar a viabilidade de um produto além de previamente identificar problemas de projeto e planejamento
MVP	Minimal Viable Product - Produto Viável Mínimo - é a versão viável de um produto que pode ser lançada com um mínimo de esforço.
UI	User Interface - é a interface do produto com o usuário
NPC	Non-Playable Character - um personagem de jogo que não é controlado por nenhum jogador.

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
1.1	MOTIVAÇÃO	16
1.2	PROBLEMA DE PESQUISA	17
1.3	METODOLOGIA	17
<b>2</b>	<b>JOGOS EDUCACIONAIS, CRIAÇÃO E VALIDAÇÃO</b>	<b>18</b>
2.1	O QUE SÃO JOGOS	18
2.2	O QUE SÃO JOGOS EDUCACIONAIS/SÉRIOS	18
<b>2.2.1</b>	<b>Classificando Jogos Educacionais/Sérios</b>	<b>19</b>
2.3	COMO CRIAR CONCEITUALMENTE JOGOS EDUCACIONAIS	21
<b>2.3.1</b>	<b>ENDO-GDC</b>	<b>22</b>
2.3.1.1	Problemas	23
2.3.1.2	Jogador/Aluno	23
2.3.1.3	Conteúdo Pedagógico	24
2.3.1.4	Objetivos de Aprendizado	24
2.3.1.5	Feedbacks Educativos	25
2.3.1.6	Inspirações	26
2.3.1.7	História	27
2.3.1.8	Objetivos do Jogo	27
2.3.1.9	Estética	27
2.3.1.10	Dinâmicas	27
2.3.1.11	Mecânicas	28
2.3.1.12	Restrições	28
<b>2.3.2</b>	<b>6-11 Framework</b>	<b>28</b>
2.4	FERRAMENTAS DE DESENVOLVIMENTO DE JOGOS	30
2.5	COMO AVALIAR JOGOS EDUCACIONAIS	31
<b>2.5.1</b>	<b>Formato do Questionário</b>	<b>32</b>
<b>2.5.2</b>	<b>Avaliação do Questionário</b>	<b>33</b>
2.6	CONCLUSÃO	34
<b>3</b>	<b>COMO ENSINAR SQL</b>	<b>35</b>
3.1	O QUE É SQL	35
3.2	QUAL A IMPORTÂNCIA DA SQL	36
3.3	COMO SQL É ENSINADO	37
3.4	JOGOS PARA ENSINAR SQL	38
3.5	CONCLUSÃO	40

<b>4</b>	<b>PROPOSTA CONCEITUAL DE UM JOGO EDUCACIONAL SOBRE SQL</b>	<b>41</b>
4.1	ENDO-GDC	41
4.1.1	Problemas	41
4.1.2	Jogador/Aluno	41
4.1.3	Conteúdo Pedagógico	41
4.1.4	Objetivos de Aprendizado	42
4.1.5	Feedbacks Educativos	42
4.1.6	Inspirações	42
4.1.7	História	43
4.1.8	Objetivos do Jogo	44
4.1.9	Estética	44
4.1.10	Dinâmicas	45
4.1.11	Mecânicas	45
4.1.12	Restrições	45
4.2	6-11 Framework	46
4.2.1	Emoções e instintos	46
4.2.2	Dinâmicas	47
4.2.3	Diagrama	47
4.3	CONCLUSÃO	47
<b>5</b>	<b>IMPLEMENTAÇÃO DE UM MVP DO SQL-GAME</b>	<b>50</b>
5.1	HISTÓRIA	50
5.1.1	A jornada do herói	50
5.1.2	A história do jogo	52
5.2	PROTOTIPAGEM	54
5.3	ESQUEMA DE FASES	56
5.4	PROVA DE CONCEITO	59
5.5	PRODUTO VIÁVEL MÍNIMO - MVP	59
5.6	CONCLUSÃO	64
<b>6</b>	<b>AVALIAÇÃO DO SQL-GAME</b>	<b>65</b>
6.1	AVALIAÇÃO	65
6.1.1	Estética	66
6.1.2	Aprendizibilidade	67
6.1.3	Operabilidade	67
6.1.4	Acessibilidade	67
6.1.5	Confiança	68
6.1.6	Desafio	68
6.1.7	Satisfação	69

<b>6.1.8</b>	<b>Interação Social</b>	69
<b>6.1.9</b>	<b>Diversão</b>	69
<b>6.1.10</b>	<b>Atenção Focada</b>	70
<b>6.1.11</b>	<b>Relevância</b>	70
<b>6.1.12</b>	<b>Aprendizagem Percebida</b>	70
<b>6.1.13</b>	<b>Objetivos de Aprendizagem</b>	71
<b>6.2</b>	<b>CONCLUSÃO</b>	71
<b>7</b>	<b>CONCLUSÃO</b>	72
	<b>REFERÊNCIAS</b>	74
	<b>APÊNDICE A – AVALIAÇÃO DISCENTE: SQLGAME, JOGO EDUCACIONAL PARA ENSINO DE SQL</b>	77
	<b>APÊNDICE B – RESPOSTAS DO FORMULÁRIO DE AVALIAÇÃO</b>	82
	<b>APÊNDICE C – TENDÊNCIA CENTRAL DE CADA PERGUNTA</b>	96
	<b>APÊNDICE D – SCRIPT PARA GERAÇÃO DAS FASES</b>	97
<b>D.1</b>	<b>FASE 1</b>	97
<b>D.2</b>	<b>FASE 2</b>	105
<b>D.3</b>	<b>FASE 3</b>	115
<b>D.4</b>	<b>FASE 4</b>	128
<b>D.5</b>	<b>FASE 5</b>	142

## 1 INTRODUÇÃO

Este projeto de final de curso apresenta um jogo educacional para o ensino de SQL, que foi testado com pessoas de diferentes áreas e níveis de conhecimento e pretende auxiliar estudantes e entusiastas de computação a terem os conhecimentos básicos sobre o assunto.

Levando em consideração que os jogos estão presentes na vida das pessoas desde sua infância até sua vida adulta/idosa (ZYDA, 2005), é natural que os mesmos acabassem sendo incorporados em outras áreas da vida além do puro entretenimento.

Os chamados Jogos Sérios (ABT, 2002) podem ser usados não só na educação, como em pesquisas, análises, na indústria e em treinamentos, isso porque eles conseguem ser sérios, sem ser solenes, interessantes sem ser hilários e difíceis sem ser frustrantes.

Eles conseguem lidar com importantes problemas comportamentais e tratar de problemas substanciais em todas as áreas acadêmicas e intelectuais (ABT, 2002), sendo uma poderosa ferramenta para aprendizado, como no caso do ensino de SQL.

Sendo assim, decidimos investigar seu uso por meio da elaboração de um jogo educativo com o objetivo de ensinar SQL para estudantes de computação.

### 1.1 MOTIVAÇÃO

Durante os anos, linguagens de programação surgem e desaparecem com frequência, mas apenas algumas continuam vivas por tempo suficiente para se tornar referência em uma dada área. Nessa situação temos linguagens como Cobol, muito utilizado ainda no desenvolvimento de mainframes, e C que ainda é muito popular para desenvolvimento de sistemas operacionais, por exemplo. Quando falamos da área de de dados, temos o SQL como esta linguagem de referência, cujas raízes remontam a 1970 (BEAULIEU, 2009).

O SQL (Structured Query Language) é uma linguagem para geração, manipulação e recuperação de dados de um banco de dados relacional. Um dos principais motivos de sua popularidade é a própria popularidade dos bancos de dados relacionais, que, quando elaborados de maneira eficiente, conseguem armazenar quantidades enormes de dados, além de permitir uma visualização de dados melhor do que muitas outras alternativas de modelo (BEAULIEU, 2009).

Por esse e outros motivos o SQL não apenas está presente na área há um bom tempo, como ainda tem um futuro brilhante a sua frente.

Não é surpresa, então, que o SQL é uma linguagem amplamente ensinada em universidades, mas acaba por ser um problema comum para muitos estudantes de computação por envolver um conjunto de passos bem diferentes àquelas ministradas ao se aprender linguagens procedurais e orientadas a objetos (GARNER; MARIANI, 2015).

## 1.2 PROBLEMA DE PESQUISA

Este projeto final tem como objetivo principal elaborar e implementar um jogo sério para ensino de SQL, de forma a proporcionar a um jogador, relativamente leigo no assunto, a teoria e prática necessária para o aprendizado dos conceitos iniciais da linguagem.

Para isso, foram definidas três questões de pesquisa que viabilizam o projeto:

- a) Como dever ser projetado um jogo educativo?
- b) Como avaliar o jogo como interessante e atraente para o jogador/aluno?
- c) Quais conteúdos didáticos devem ser abordados ao ensinar SQL?

A partir das respostas, teóricas, às três questões, foi possível então construir um Produto Viável Mínimo (TAUCEI, 2019) para o jogo em questão e avaliá-lo apropriadamente.

## 1.3 METODOLOGIA

A especificação do problema de pesquisa levou a definição das seguintes questões de pesquisa:

**Q1** O que define um jogo?

**Q2** O que é um jogo educativo?

**Q3** Como projetar o *game design* de um jogo educativo?

**Q4** Por que ensinar SQL?

**Q5** Quais conteúdos didáticos de SQL o jogo deve cobrir?

**Q6** Como implementar o jogo?

**Q7** Como avaliar o jogo?

Para gerenciamento do processo de desenvolvimento do jogo, utilizamos a ferramenta Trello, uma ferramenta visual que possibilita ao time o gerenciamento de qualquer tipo de projeto, fluxo de trabalho ou monitoramento de tarefas (ATLASSIAN, 2022) e que tem a organização semelhante a de um *kanban*.

Para planejamento do jogo utilizamos o Endo-GDC, a ser explicado em detalhe no capítulo 2, mas que se trata de um *Game Design Canvas* para jogos educativos endógenos (TAUCEI, 2019).

E por fim, para prototipação e implementação do jogo utilizamos, respectivamente, o Figma, um editor gráfico de vetor e prototipagem de projetos de design, e a Unity, um motor de jogo proprietário criado pela Unity Technologies.

## 2 JOGOS EDUCACIONAIS, CRIAÇÃO E VALIDAÇÃO

### 2.1 O QUE SÃO JOGOS

Pode parecer trivial discutir esta definição, mas na verdade é uma tarefa extremamente complexa. Ao longo do tempo, diversas definições foram propostas, todas tentando tornar mais claro um conceito desfocado (XEXEO et al., 2017).

Não há nenhuma definição precisa na literatura, e sim várias que são aceitas pelo menos como parcialmente certas e, no dia a dia, todas conseguem identificar um jogo.

Uma definição proposta pelo laboratório LUDES da UFRJ (XEXEO et al., 2017) do que seria um jogo é:

Jogos são atividades sociais e culturais voluntárias, significativas, fortemente absorventes, não produtivas, que se utilizam de um mundo abstrato, com efeitos negociados no mundo real, e cujo desenvolvimento e resultado final é incerto, onde um ou mais jogadores, ou equipes de jogadores, modificam interativamente e de forma quantificável o estado de um sistema artificial, possivelmente em busca de objetivos conflitantes, por meio de decisões e ações, algumas com a capacidade de atrapalhar o adversário, sendo todo o processo regulado, orientado e limitado, por regras aceitas, e obtendo, com isso, uma recompensa psicológica, normalmente na forma de diversão, entretenimento, ou sensação de vitória sobre um adversário ou desafio.

Para MCGONIGAL e RIECHE (2012), os jogos compartilham de quatro características que os definem.

- a) Meta: o objetivo que eles pretendem alcançar no jogo.
- b) Regras: são regras que restringem as ações do jogador, dificultando que ele alcance a Meta.
- c) Sistema de feedback: têm como função informar o jogador quão perto ele está de atingir a meta, sendo também uma forma de incentivo.
- d) Participação voluntária: os jogadores devem aceitar de forma consciente e voluntária as regras e feedbacks.

### 2.2 O QUE SÃO JOGOS EDUCACIONAIS/SÉRIOS

Os jogos educacionais, também chamados de "Jogos Sérios", têm seu significado explorado por diversos autores. Xexeo et al. (2017) afirma que os Jogos Sérios poderiam ser vistos como uma nova classe, que compartilha comportamentos com a classe de Jogos propriamente dita, mas são produtivos.

---

<sup>1</sup> <<http://ludes.cos.ufrj.br>>

Xexeo et al. (2017) conclui que a definição mais comum encontrada na literatura sobre Jogos Sérios é a de que é um tipo de jogo criado com o objetivo principal de não ser somente focado no entretenimento, mas sem abrir mão do mesmo.

O conceito de jogos sérios também foi explorado por Abt (2002), que alegava que tais jogos possuíam um propósito educacional explícito e bem definido, não sendo meramente formas de lazer ou diversão.

Para Susi, Johannesson e Backlund (2007) os jogos sérios têm como objetivo principal treinar ou educar o usuário, apesar de poder ter outros propósitos como propagandas ou marketing, enquanto lhe proporciona uma experiência agradável.

Alves (2013) salienta que jogos sérios são jogos que não têm como principal objetivo a diversão ou o entretenimento, o que está de acordo com a definição acima. Além disso, afirma que tais jogos são assim chamados de sérios, não porque os outros não o são, mas sim porque estes são usados de um forma pedagógica para fins políticos, sociais, publicitários, econômicos, ambientais ou causas humanitárias.

Olhando o quadro 1, é possível ter um entendimento maior da diferença entre jogos e jogos sérios, com uma visão comparativa das diferenças entre esses dois conceitos na prática.

Quadro 1 – Diferenças entre jogos de entretenimento e jogos sérios

Critério	Jogos Sérios	Jogos de Entretenimento
Tarefa vs Experiência Rica	Foco em resolução de problemas	Preferência em proporcionar experiências ricas
Objetivo	Elementos importantes de aprendizado	Se divertir
Simulações	Suposições necessárias para simulações viáveis	Processo de simulação simplificado
Comunicação	Deve refletir a comunicação natural (não perfeita)	Comunicação normalmente é perfeita

Fonte: Susi, Johannesson e Backlund (2007)

### 2.2.1 Classificando Jogos Educacionais/Sérios

Tal como os jogos, o termo "jogos sérios", por si só, representa um conceito complexo que pode ser subdividido em categorias menores. Alves (2013) explora as classificações existentes dos jogos sérios que, apesar de ainda não estarem muito sólidas, possuem termos comuns para alguns dos tipos existentes.

É possível ver a lista completa das principais classificações no quadro 2. Dentre eles, o que diz respeito ao jogo que estará sendo elaborado nessa dissertação é o *Edutainment*: jogo de entretenimento que é projetado para ser educativo.

Quadro 2 – Tipos de jogos sérios

Designação (inglês)	Descrição
Advergames	Jogo utilizado para promover uma marca, um produto, uma organização ou um ponto de vista.
Edutainment	Jogo de entretenimento que é projetado para ser educativo.
Game-based Learning	Jogo com objetivos de aprendizagem. São projetados de forma a equilibrar a componente lúdica com a componente didática.
Newsgames	Jogo jornalístico que reporta acontecimentos recentes ou envia comentários editoriais sobre o acontecimento.
Training and Simulation Games	Jogo que simula atividades da vida real com a maior exatidão possível. Utilizado para adquirir ou exercitar diferentes habilidades.
Persuasive Games	Jogo que influencia os jogadores a agirem através da jogabilidade. É projetado para mudar atitudes e comportamentos dos utilizadores através da persuasão.
Organizational-dynamic	Jogo projetado com propósitos específicos de promover o desenvolvimento pessoal e formação de caráter. Aborda questões de dinâmica organizacional, em três níveis: comportamento individual, dinâmica de grupo e rede e dinâmica cultural.
Games for Health	Jogo utilizado para ensaios médicos, educação de saúde, terapia psicológica e reabilitação física ou cognitiva.
Art Games	Jogo utilizado para expressar ideias artísticas ou arte produzida através de jogos digitais.
Militainment	Jogo utilizado para fins militares que replica operações com um alto grau de precisão.

Fonte: Alves (2013)

Indo ainda mais a fundo da classificação do jogo fruto dessa dissertação, focaremos nos jogos de computador educativos (EGENFELDT-NIELSEN, 2011), ou seja, jogos de computador desenvolvidos para fins educacionais.

Sendo assim, jogos sérios, jogos educacionais e jogos de computador educativos são considerados sinônimos nesta dissertação.

Portanto, para esta dissertação, os jogos educativos devem possuir metas, regras e sistemas de feedback, conforme definido por [MCGONIGAL e RIECHE \(2012\)](#), devem engajar o usuário [\(SUSI; JOHANNESSON; BACKLUND 2007\)](#) e possuir um propósito educacional explícito [\(ABT 2002\)](#).

### 2.3 COMO CRIAR CONCEITUALMENTE JOGOS EDUCACIONAIS

Os jogos sérios são uma excelente estratégia educacional pois permitem que os jogadores vivenciem situações que são impossíveis de serem vivenciadas no mundo real [\(TAUCEI 2019\)](#) por questões de custo, segurança, tempo, etc.

De acordo com [Eck \(2006\)](#), os jogos sérios incorporam todas as fases que ativam os processos necessários para uma aprendizagem efetiva, os quais estão presentes nos nove eventos de [Gagné, Wagner e Golas \(2004\)](#), que são: ganhar a atenção, informar os objetivos do aluno, estimular a recordação de aprendizagem anterior, apresentar material de estímulo, fornecer orientação ao aluno, provocar o desempenho, fornecer feedback, avaliar o desempenho e aprimorar a retenção e a transparência.

Já [Lieberman \(2006\)](#) lista alguns benefícios do uso de jogos sérios para fins educacionais:

- a) Os jogos proporcionam ao jogador uma experiência ativa;
- b) Jogos encorajam o jogador a aprender fazendo;
- c) Os jogos são um meio social que proporciona ao jogador interações entre pessoas e respostas emocionais;
- d) Os jogos são participativos, fornecendo ao jogador feedback personalizado e rápido;
- e) Jogos são envolventes. A participação faz com que o jogador preste muita atenção, além de exigir um planejamento e tomada de decisão cuidadosa, exige também a aprendizagem para ter sucesso;
- f) Jogos promovem a aprendizagem comportamental. O jogo dá ao jogador recompensas por comportamento (pontos, poder, classificação e assim por diante). Esse feedback positivo no jogo pode encorajar comportamentos desejados na vida real;
- g) Jogos oferecem consequências. Estes não são abstratos ou hipotéticos; eles são representados no jogo diretamente. O jogador interpreta um personagem e se identifica com ele ou ela. Sucesso e fracasso mapeiam diretamente as suas ações; o ego e a auto-imagem são investidos na experiência;
- h) Jogos fornecem modelos para o jogador. O jogador pode aprender com os personagens do jogo e entender suas experiências comportamentais.

Dessa forma, entende-se que o jogo oferece oportunidades para melhoria de aprendizado mas apenas se ele for desenvolvido de maneira eficaz [\(WINN 2009\)](#). Segundo a

autora, fazer um jogo, por si só, já é uma tarefa difícil pois um bom jogo requer um processo de design que, além de criativo, às vezes é um pouco caótico. Quando falamos então sobre o desenvolvimento de um jogo educacional, a maior dificuldade é que, em vez de simplesmente tentar otimizar o aspecto de entretenimento do jogo, também é preciso otimizá-lo para alcançar um conjunto específico de resultados de aprendizado.

Mas, assim como acontece no desenvolvimento de jogos de entretenimento, o desenvolvimento de jogos sérios precisa de profissionais de diferentes especialidades. A diferença principal, entretanto, é a necessidade de incluir um especialista na área que será lecionada, além de especialidades envolvidas com a área de educação, como a pedagogia (TAUCEI, 2019).

Sendo assim, uma equipe de desenvolvimento de jogos educacionais acaba tendo um caráter multidisciplinar.

Existem 3 fatores que favorecem o desenvolvimento colaborativo de um jogo sério: respeitar a especialidade de cada membro da equipe, participar do desenvolvimento iterativo e ter uma visão compartilhada do produto do jogo (TAUCEI, 2019). Segundo o autor, uma ferramenta que pode ajudar nesses dois últimos itens é a utilização de um *canvas*.

Ainda segundo Taucei (2019), um *canvas* é um quadro ou um painel que ajuda a sintetizar e organizar as informações sobre o artefato que se deseja desenvolver, seja ele: um modelo de negócio, um produto, experiência do usuário, jogos, etc. É uma ferramenta que tem como objetivo dar uma visão do todo, permitindo a identificação de lacunas no projeto, mantendo o foco no que é mais relevante.

Com isso em mente, Taucei (2019) em sua dissertação explora os *canvas* para desenvolvimentos de jogos, os *Game Design Canvas* (GDCs) e suas principais referências para, por fim, propor um novo GDC para jogos endógenos sérios focados no aprendizado cognitivo: o ENDO-GDC.

### 2.3.1 ENDO-GDC

Um jogo endógeno, de um modo geral, foca em objetivos de aprendizagem mais complexos, integrando o conteúdo de aprendizagem melhor à estrutura do jogo (TAUCEI, 2019).

O ENDO-GDC pode ser visto na Figura 1. Este é formado por seis blocos que estão ligados às camadas que compõe um jogo sério (TAUCEI, 2019), tais são:

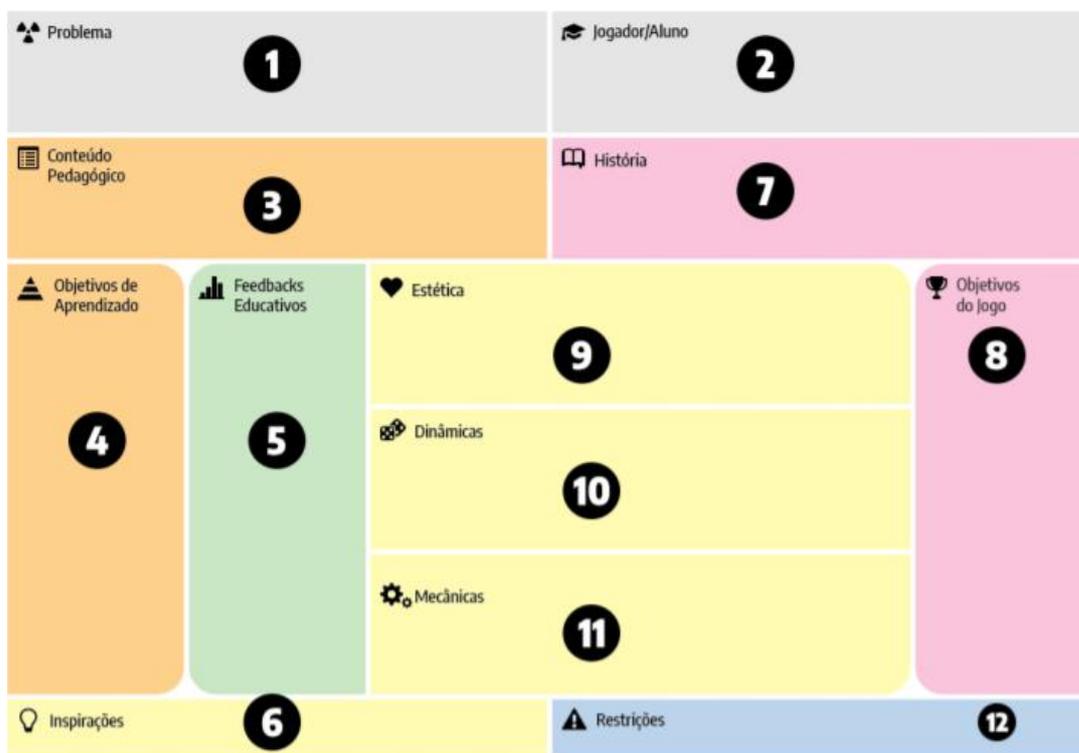
**bloco do jogador** (cinza): com as seções problema e jogador/aluno, em cinza

**bloco aprendizado** (laranja): com as seções conteúdo pedagógico e objetivos de aprendizado, em laranja

**bloco narrativa** (rosa): com as seções história do jogo e objetivos do jogo, em rosa

**bloco gameplay** (amarelo): com as seções mecânicas, dinâmicas e estética, em amarelo

Figura 1 – Modelo ENDO-GDC



Fonte: Taucei (2019)

**bloco experiência do usuário** (verde): com a seção feedback em verde

**bloco tecnologia** (azul): formada pelas seções tecnologia e plataforma em azul

São um total de 12 campos que devem ser preenchidos na ordem indicada na figura 1 para melhor elaboração do jogo (TAUCEI, 2019).

### 2.3.1.1 Problemas

Esta seção procura descrever os problemas que se espera solucionar com o jogo, podendo estes surgir das áreas do conhecimento que podem gerar dificuldade ou de problemas relativos a falta de motivação para aprender dos jogadores/alunos (TAUCEI, 2019).

### 2.3.1.2 Jogador/Aluno

Os Jogadores são o motivo de o jogo existir então é natural que seja um dos primeiros parâmetros a ser determinado no preenchimento do *canvas*.

A experiência do jogo como um todo é influenciada não apenas pelo *game design*, como também pelo *background* cognitivo, social, cultural e experiencial que o jogador traz ao jogar (WINN, 2009). Sendo assim, entender o jogador é uma etapa chave para desenvolver o jogo.

Segundo Taucei (2019) essa seção descreve quem é o público alvo do jogo, ou seja, o conjunto de pessoas que irão jogar e aprender através da *gameplay*. Para isso, alguns dados são importantes: faixa etária, profissão, gênero, nível escolar e cenário de uso (TAUCEI, 2019).

### 2.3.1.3 Conteúdo Pedagógico

Segundo Taucei (2019), O conteúdo pedagógico é a listagem de assuntos e conceitos que serão abordados pelo jogo, devendo se manter coerentes com o público alvo definido na seção Aluno/Jogador. Dessa forma, devem ser listados:

- a) Área de conhecimento.
- b) Tópicos da área de conhecimento que serão abordados no jogo.
- c) Conteúdo que se deseja ensinar e que está relacionado com a área de conhecimento.

Esta seção é importante pois servirá como guia para a elaboração dos objetivos de aprendizado baseados no conteúdo pedagógico definido (TAUCEI, 2019).

### 2.3.1.4 Objetivos de Aprendizado

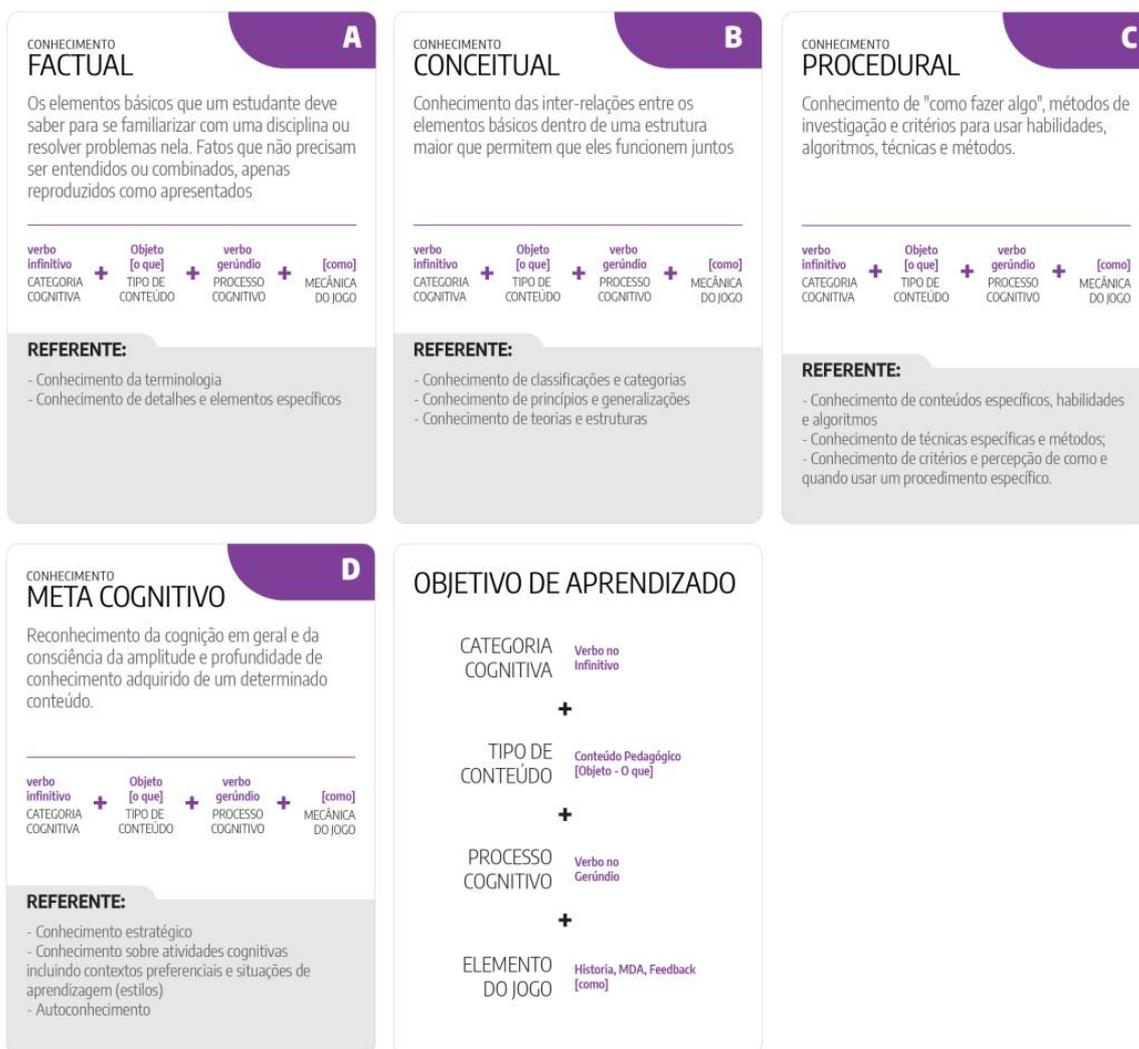
Segundo Taucei (2019), esta seção é importante para conseguir determinar se o jogo atingiu o seu propósito educativo, que deve ir além do puro entretenimento do jogador. Além disso, servirá como um guia para a construção do jogo, uma vez que as mecânicas, dinâmicas, estéticas e o *feedback* que devem ser desenvolvidas pensando em como alcançar esses objetivos.

Nessa seção é descrito o que se espera que o Jogador/Aluno aprenda ao jogar o jogo, dentro das três categorias amplas de resultado de aprendizado: cognitivo (focado no conhecimento), afetivo (focado nas emoções e sentimentos) e psicomotor (focado nas habilidades motoras) (TAUCEI, 2019).

Para elaboração dos Objetivos de Aprendizado, Taucei (2019) elaborou as Cartas de Bloom, baseadas na Taxonomia de Bloom, que é uma ferramenta pedagógica usada para classificar os objetos de aprendizagem educacional em níveis de complexidade e especificidade.

As cartas de Domínio de Conhecimento foram organizadas de acordo com o nível de abstração do conhecimento, indo do mais concreto 'A', ao mais abstrato 'D' (TAUCEI, 2019). Na figura 2 é possível ver cada dimensão do conhecimento com sua definição, tipo de conhecimento do qual se refere (na parte mais inferior da carta), formato de sentença a ser formada (meio da carta), que é constituída pelo sujeito (quem vai aprender), o verbo do processo cognitivo (cartas de processo cognitivo que podem ser visto na figura 3) e o objeto que é o tipo de conteúdo ou conhecimento que será utilizado (TAUCEI, 2019). Este último é formado, normalmente, por cada um dos itens listados na seção de conteúdo pedagógico (TAUCEI, 2019).

Figura 2 – Cartas de Dimensão de Conhecimento



Fonte: Taucei (2019)

A dimensão dos processos cognitivos está dividida nas 6 cartas da figura 3, cada uma representando um processo cognitivo, do menos complexo (carta "Lembrar" de número "1") ao mais complexo (carta "Criar" de número "06") (TAUCEI 2019).

Cada um dos processos cognitivos tem a sua definição descritos nas cartas, além de listar os verbos que representam o processo cognitivo em questão.

### 2.3.1.5 Feedbacks Educativos

Segundo Taucei (2019), o *feedback* é muito importante pois diz ao jogador quão perto ele está de atingir os objetivos do jogo, sendo uma das principais formas de manter o jogador motivado.

Essa seção busca descrever as informações ou dados fornecidos ao jogador quando este

Figura 3 – Cartas de Processo Cognitivo



Fonte: Taucei (2019)

interage com o jogo, sendo focados na motivação do jogador e no seu aprendizado. Eles devem ser descritos com o nome como título do post-it, o que faz, e qual mecânica e objetivo pedagógico está associado quando este for relacionado ao aprendizado.

Os *feedback* podem ser analisados no *playtest* para saber se estão auxiliando no processo de *debriefing* do jogador.

### 2.3.1.6 Inspirações

Esta seção tem como objetivo reunir referências de jogos que podem servir de inspiração para o jogo em questão. Pesquisar sobre o que já deu certo ou errado no desenvolvimento de outros jogos pode ajudar a elaborar o jogo de uma maneira mais completa e diferenciada (TAUCEI, 2019).

Nessa seção, para cada jogo referenciado, devem ser feitas as seguintes perguntas (TAUCEI, 2019):

- a) Quais elementos do jogo tornam sua experiência agradável?
- b) Que elementos do jogo tornam sua experiência desagradável?
- c) Como posso mudar os elementos do jogo para melhorar a experiência?

### 2.3.1.7 História

A história do jogo é a sequência de eventos que se desdobram nele, existindo dois tipos de histórias: a do designer e a do jogador. O *canvas* em questão foca na história do designer, que é responsável por fornecer propósito, engajamento, definir cenários e transmitir o conteúdo, além de ser um meio de transmitir o conteúdo pedagógico (TAUCEI, 2019).

### 2.3.1.8 Objetivos do Jogo

De acordo com Taucei (2019), o objetivo é importante pois fornece um propósito ao jogador, indicando o que ele deve fazer para ganhar o jogo. Objetivos claros, difíceis e específicos, levam a um aumento de desempenho do usuário, além de serem cruciais no desencadeamento de maior atenção e motivação.

Dessa forma, segundo Taucei (2019), esta seção deve ser preenchida com os objetivos que o jogador deve alcançar no jogo. Estes objetivos devem, é claro, estar alinhados com a história do jogo, e, na medida do possível, estar alinhados com os objetivos de aprendizado. Devem ser desenvolvidos, então *feedbacks* para eles, que informem o jogador como está seu desempenho em relação a esses objetivos.

### 2.3.1.9 Estética

As estéticas descrevem as respostas emocionais que se deseja evocar no jogador e os elementos que dão aparência sensorial ao jogo. Além disso, a estética dá forma ao *feedback* e informa o jogador sobre o seu desempenho e funcionamento do mundo e tem que ser coerente com o mundo definido pela história.

Nesta seção devem ser definidas as emoções que o jogo deseja despertar no jogador, assim como deve ser definido o design geral do jogo (TAUCEI, 2019).

### 2.3.1.10 Dinâmicas

Esta seção deve descrever as dinâmicas do jogo. Elas são a ponte entre a estética e a mecânica, descrevendo o comportamento em tempo de execução da mecânica que atua sobre as entradas do jogador a as outras saídas ao longo do tempo (TAUCEI, 2019).

### 2.3.1.11 Mecânicas

De acordo com Taucei (2019), as mecânicas são cruciais para o *gameplay* e por isso são o núcleo, a essência de um jogo. Elas estabelecem os procedimentos e regras que o jogador deve seguir para interagir com o jogo, ou seja, informam como o jogador poderá alcançar os objetivos.

Aqui devem ser definidas as principais mecânicas do jogo, relacionando-as com os objetivos de aprendizado de forma a possibilitar o desenvolvimento de um jogo endógeno. Essa combinação de conteúdo pedagógico com as mecânicas permite que o jogador aprenda através do *gameplay* (TAUCEI 2019),

### 2.3.1.12 Restrições

O último elemento a ser preenchido no *canvas*, as Restrição especificam a tecnologia que será utilizada para o desenvolvimento do jogo e a plataforma onde este jogo irá rodar, podendo ser analógico (jogos de carta, por exemplo), ou digital (celulares, PC, consoles, etc) (TAUCEI, 2019).

Este deve ser o último elemento a ser preenchido pois é fácil se deixar levar pelas possibilidades que a tecnologia pode fazer e se esquecer que o propósito é criar um ótimo jogo (TAUCEI 2019).

## 2.3.2 6-11 Framework

Além da montagem do ENDO-GDC, como uma forma de entender e avaliar os sentimentos que as dinâmicas e mecânicas do jogo causam no jogador, pode ser usado o 6-11 *Framework*, uma estrutura que permite que desenvolvedores de jogos consigam engajar emocionalmente e divertir seus jogadores (DILLON 2011).

O 6-11 *Framework* é baseado no modelo MDA (*Mechanics, Dynamics e Aesthetics*), três aspectos também explorados no ENDO-GDC. O *framework* tenta auxiliar na especificação e entendimento das Estéticas criando uma nova taxonomia que consegue se relacionar mais facilmente com as dinâmicas (DILLON 2011).

Dillon (2011) alega que jogos podem ser tão envolventes em um nível subconsciente porque eles conseguem com sucesso se basear em um subconjunto de emoções básicas e instintos, os quais são comuns e estão profundamente enraizados em todos nós.

Dessa forma, o *framework* foca em seis emoções e onze instintos que são recorrentes na psicologia.

As seis emoções descritas por Dillon (2011) são:

**Medo:** um dos sentimentos mais comuns nos jogos cotidianos, pensando em todos os mais recentes jogos de terror e sobrevivência, ou de explorações de *dungeons* e mundos desconhecidos.

**Raiva:** um sentimento poderoso que pode ser um fator motivacional para o jogador jogar de novo ou avançar na história.

**Alegria:** um dos sentimentos mais relevantes para o divertimento do jogo. Normalmente é a consequência do jogador vencer algum desafio ou ganhar alguma recompensa.

**Orgulho:** recompensando o jogador e o fazendo se sentir bem por suas conquistas é um fator motivacional importante para os motivar a avançar no jogo e enfrentar desafios ainda mais difíceis.

**Tristeza:** apesar de ser uma emoção que não parece combinar muito com o objetivo de “diversão”, é uma forma de atingir novos objetivos artísticos e abordar temas mais complexos e maduros.

**Excitação:** os jogos que se vale a pena jogar devem atingir esse sentimento de maneira natural como uma consequência de ativar com sucesso outras emoções ou instintos.

Enquanto que os onze instintos explorados por [Dillon \(2011\)](#) são:

**Sobrevivência (Luta ou Fuga):** o mais fundamental e primordial de todos os instintos, desencadeada quando nós, como qualquer outro ser, nos sentimos ameaçados. Diante de uma situação de perigo, nosso cérebro vai instantaneamente decidir se devemos enfrentar a ameaça e lutar por nossa vida ou tentar evitá-la encontrar uma possível maneira de escapar. Isto é amplamente utilizado em muitos videogames modernos.

**Autoidentificação:** as pessoas tendem a admirar indivíduos bem-sucedidos ou personagens fictícios inteligentes e naturalmente começam a se imaginar sendo como eles.

**Colecionar:** um instinto muito forte que pode ligar a uma variedade de emoções diferentes e sempre foi muito usado em jogos.

**Ganância:** muitas vezes estamos propensos a ir além de uma simples “coleção” e começamos a acumular muito mais do que é realmente necessário apenas por causa disso. Se estamos falando de itens verdadeiramente valiosos ou apenas bens e recursos que precisamos para construir nosso império virtual em um jogo de estratégia, é provável que um instinto ganancioso apareça bem cedo em nossos hábitos de jogo.

**Proteção/Cuidado:** indiscutivelmente o “melhor” instinto de todos: aquele que faz com que todos os pais amem seus filhos e faz com que as pessoa sintam o impulso de cuidar e ajudar os necessitados.

**Agressividade:** o outro lado da moeda, geralmente levando à violência quando acoplado com ganância ou raiva. É explorado em inúmeros jogos.

**Vingança:** outro instinto poderoso que pode agir como uma força motivacional e é frequentemente usado em jogos para avançar o enredo ou justificar porquê precisamos aniquilar um cara mau.

**Competição:** profundamente ligada aos aspectos da nossa psique e um dos instintos mais importantes em relação ao jogo, por exemplo, tabelas de classificação. Sem isso, os jogos perderiam muito do seu apelo.

**Comunicação:** a necessidade de expressar ideias, pensamentos, ou apenas fofocas, foi um dos mais influentes para a evolução humana e pode ser usado com grande efeito em jogos também, enquanto busca informações conversando com um personagem não jogável (NPC) ou enquanto compartilhar experiências com outros jogadores em chats e fóruns.

**Curiosidade:** todas as descobertas humanas, sejam científicas ou de natureza geográfica, foram feitas graças a esses instintos que sempre nos empurraram em direção ao desconhecido.

**Apreciação de cores:** cenas e ambientes cheios de cores vibrantes naturalmente nos atraem, incluindo cada vez mais gráficos detalhados e coloridos que vemos em jogos modernos.

## 2.4 FERRAMENTAS DE DESENVOLVIMENTO DE JOGOS

No mercado de jogos atual, existem tantas *game engines* que é difícil saber ao certo qual escolher para desenvolver um jogo, Tyler (2022), por exemplo, lista 25 das *engines* mais populares do mercado.

Uma *game engine*, de acordo com Lewis e Jacobson (2002), é uma coleção de módulos de simulação de código que não diretamente especifica o comportamento do jogo (sua lógica), ou o ambiente do jogo. As *engines* abrangem módulos que lidam com *input*, *output* (renderização 3D, desenho 2D, som, etc), e físicas e dinâmicas genéricas do mundo dos jogos.

Analisando as listas das *engines* mais populares como a feita por Lima (2018), ou a feita por Tyler (2022) que é bem mais recente, algumas *engines* aparecem nas duas listas de maneira consensual, tais são:

- a) Unreal Engine: é de longe a *engine* que tem maior poder gráfico da atualidade, permitindo criação de jogos 3D com maior qualidade gráfica e foto-realismo (LIMA, 2018).

Muitos jogos famosos foram feitos usando essa *engine*, tais como a série de jogos Gears of War, a série de jogos Mass Effect e a série de jogos Batman: Arkham (TYLER, 2022).

- b) Unity: é uma das mais populares e a principal concorrente da Unreal Engine. É muito conhecida por ser completa e bem amigável para iniciantes (LIMA, 2018).

Muitos desenvolvedores *indies* usam a Unity para desenvolver seu jogos, por ela se mostrar bem funcional e com conteúdos de alta qualidade, além de sua habilidade de ser usado para basicamente qualquer jogo (TYLER, 2022).

Há também jogos famosos que foram feitos usando a Unity Engine, tais como Lara Croft Go, Her Story e Pillars of Eternity.

- c) Game Maker: É uma das *engines* mais antigas do mercado, mas se mostrando ainda, ao mesmo tempo, poderosa e simples (LIMA, 2018). Ela se popularizou bastante por não exigir conhecimentos em programação para o desenvolvimento do jogo, onde o desenvolvedor usava mecânicas de "clique" e "arrasto" para definir todas as mecânicas e regras do jogo.

Mas uma das principais desvantagens dessa *engine* é que ela não é gratuita o que acaba afastando muitos desenvolvedores iniciantes ou amadores dela (LIMA, 2018).

Alguns dos jogos mais famosos feitos usando o Game Maker são: Spelunky, Hotline Miami, Super Crate Box (TYLER, 2022) e Undertale (LIMA, 2018).

As três *engines*, no quesito de conhecimento prévio, estavam equiparadas. Entretanto nosso conhecimento na Unity Engine era consideravelmente superior às demais, o que justificou a escolha.

A Unity em si é bem completa mesmo para o desenvolvimento de jogos 3D, mas como o nosso jogo foi projetado para ser 2D, ela atendia com folga quaisquer requisitos que tivéssemos para a implementação.

## 2.5 COMO AVALIAR JOGOS EDUCACIONAIS

Nos últimos anos, os jogos vêm cada vez mais sendo usados como uma estratégia instrucional em diversas áreas de conhecimento, tais como: matemática, computação, nutrição, saúde, entre outras (PETRI; CHRISTIANE; BORGATTO, 2019).

Apesar de se acreditar que seja uma estratégia educacional eficaz e eficiente, na prática os benefícios dos jogos usados para o ensino ainda são questionáveis devido à falta de estudos que forneçam evidências desses benefícios (PETRI; CHRISTIANE; BORGATTO, 2019). Uma razão para isso é a irregularidade com que esses jogos são avaliados, muitas vezes não utilizando critérios muito ricos ou categorizáveis (PETRI; CHRISTIANE; BORGATTO, 2019).

Diante desse cenário surgiu o MEEGA+, um modelo sistemático para avaliação de jogos que tem como objetivo avaliar a percepção da qualidade do ponto de vista de alunos de cursos superiores da área de computação (PETRI; CHRISTIANE; BORGATTO, 2019).

## 2.5.1 Formato do Questionário

Petri, Christiane e Borgatto (2019) definem um conjunto de itens para o questionário do modelo MEEGA+ para cada dimensão/subdimensão, que pode ser visto no Quadro 3.

Quadro 3 – Definição das dimensões/subdimensões

Dimensão/Subdimensão		Item	Descrição do Item
Usabilidade	Estética	1	O design do jogo é atraente (interface, gráficos, tabuleiro, cartas, etc.).
		2	Os textos, cores e fontes combinam e são consistentes.
	Aprendizagem	3	Eu precisei aprender poucas coisas para poder começar a jogar o jogo.
		4	Aprender a jogar este jogo foi fácil para mim.
		5	Eu acho que a maioria das pessoas aprenderiam a jogar este jogo rapidamente.
	Operabilidade	6	Eu considero que o jogo é fácil de jogar.
		7	As regras do jogo são claras e compreensíveis.
	Acessibilidade	8	As fontes (tamanho e estilo) utilizadas no jogo são legíveis.
		9	As cores utilizadas no jogo são compreensíveis.
		10	O jogo permite personalizar a aparência (fonte e/ou cor) conforme a minha necessidade.
	Proteção contra erros do usuário	11	O jogo me protege de cometer erros.
		12	Quando eu cometo um erro é fácil de me recuperar rapidamente.
Confiança	13	Quando olhei pela primeira vez o jogo, eu tive a impressão de que seria fácil para mim.	
	14	A organização do conteúdo me ajudou a estar confiante de que eu iria aprender com este jogo.	
Desafio	15	Este jogo é adequadamente desafiador para mim.	
	16	O jogo oferece novos desafios (oferece novos obstáculos, situações ou variações) com um ritmo adequado.	
	17	O jogo não se torna monótono nas suas tarefas (repetitivo ou com tarefas chatas).	
Satisfação	18	Completar as tarefas do jogo me deu um sentimento de realização.	
	19	É devido ao meu esforço pessoal que eu consigo avançar no jogo.	
	20	Me sinto satisfeito com as coisas que aprendi no jogo.	
	21	Eu recomendaria este jogo para meus colegas.	
Interação Social	22	Eu pude interagir com outras pessoas durante o jogo.	
	23	O jogo promove momentos de cooperação e/ou competição entre os jogadores.	
	24	Eu me senti bem interagindo com outras pessoas durante o jogo.	
Diversão	25	Eu me diverti com o jogo.	
	26	Aconteceu alguma situação durante o jogo (elementos do jogo, competição, etc.) que me fez sorrir.	
Atenção Focada	27	Houve algo interessante no início do jogo que capturou minha atenção.	
	28	Eu estava tão envolvido no jogo que eu perdi a noção do tempo.	
	29	Eu esqueci sobre o ambiente ao meu redor enquanto jogava este jogo.	
Relevância	30	O conteúdo do jogo é relevante para os meus interesses.	
	31	É claro para mim como o conteúdo do jogo está relacionado com a disciplina.	
	32	O jogo é um método de ensino adequado para esta disciplina.	
	33	Eu prefiro aprender com este jogo do que de outra forma (outro método de ensino).	
Aprendizagem Recebida	34	O jogo contribuiu para a minha aprendizagem na disciplina.	
	35	O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina.	

Fonte: Petri, Christiane e Borgatto (2019)

As 35 perguntas vistas no Quadro 3 são genéricas e aplicáveis a quaisquer jogos digitais,

mas ainda há a necessidade de determinar as perguntas de conteúdo pedagógico específico no formato “O jogo contribuiu para <verbo conforme nível do objetivo de aprendizagem (cognitivo, psicomotor, afetivo)><objetivo/conceito>.” seguindo a mesma Taxonomia de Bloom que foi utilizada para determinação dos objetivos de aprendizado na seção 2.3.1.4

Todas as perguntas definidas no questionário do aluno devem ser respondidas no formato da escala Likert de 5 pontos com alternativas de respostas que variam de Discordo Totalmente a Concordo Totalmente (PETRI; CHRISTIANE; BORGATTO, 2019). O uso dessa escala permite expressar a opinião do indivíduo (aluno) sobre o objeto de estudo (jogo educativo) com maior precisão, além de permitir que o indivíduo se sinta mais confortável ao dar a sua opinião, tendo a possibilidade de escolher um ponto neutro.

Além das perguntas na escala Likert, há ainda quatro perguntas sobre informações demográficas que ajudam a entender as respostas por perfil de aluno, tais são faixa etária, sexo, frequência com que costuma jogar jogos digitais e frequência com que costuma jogar jogos não-digitais (de cartas, tabuleiro, etc.).

E, por fim, há três perguntas de preenchimento livre e opcional que ajudam a entender mais a fundo os motivos da avaliação do indivíduo, que são:

- a) O que você mais gostou no jogo?
- b) O que poderia ser melhorado no jogo?
- c) Gostaria de fazer mais algum comentário?

## 2.5.2 Avaliação do Questionário

De acordo com Petri, Christiane e Borgatto (2019), de modo a alcançar os objetivos da avaliação definido no modelo MEEGA+, os dados coletados são analisados por meio de métodos de estatística descritiva, que é definido por Petri, Christiane e Borgatto (2019) como:

Métodos estatísticos descritivos são usados para descrever e apresentar graficamente aspectos interessantes dos dados coletados (Wohlin et al., 2012). Assim, os dados coletados são analisados em termos de medidas de tendência central (mediana, média e frequência de respostas) e visualização gráfica (gráficos de frequência).

Como todas essas análises estatísticas são disponibilizadas pela plataforma de Formulários da Google, o questionário e as análises estatísticas descritivas podem ser feitas pela mesma. Alternativamente, Petri, Christiane e Borgatto (2019) disponibiliza uma planilha Excel com as fórmulas necessárias para a montagem desses gráficos e para o cálculo de tendência central.

## 2.6 CONCLUSÃO

Neste capítulo definimos o que são jogos e jogos educativos, o que clareia muito mais a visão do produto final deste projeto.

Foi possível também entender como projetar um jogo educativo usando o ENDO-GDC, um *canvas* desenvolvido por Taucei (2019) para desenvolvimento de jogos educativos endógenos em conjunto com o 6-11 *Framework*. Dessa forma, temos um embasamento muito maior no desenvolvimento do *game design* do jogo, conseguindo garantir que o produto final atenda às expectativas.

Para avaliação do jogo, foi explorada a solução desenvolvida por Petri, Christiane e Borgatto (2019), o MEEGA+, que melhora o antigo modelo MEEGA, adaptando-o para avaliar jogos educativos de computação.

Com isso as seguintes questões de pesquisa foram respondidas neste capítulo:

**Q1** O que define um jogo?

**Q2** O que é um jogo educativo?

**Q3** Como projetar o *game design* de um jogo educativo?

**Q7** Como avaliar o jogo?

Além disso, vimos um pouco sobre a sexta questão de pesquisa: “(Q6) Como implementar o jogo?”, enumerando as possíveis ferramentas (*engines*) mais populares do mercado e selecionando a que mais atendia às necessidades do projeto.

### 3 COMO ENSINAR SQL

#### 3.1 O QUE É SQL

Em 1970, o Dr. E. F. Codd, do laboratório de pesquisa da IBM, publicou um artigo chamado *A Relational Model of Data for Large Shared Data Banks*, que descreve o modelo relacional, o qual propõe que os dados sejam representados como um conjunto de tabelas (BEAULIEU, 2009).

Junto com a definição do modelo relacional, Codd propôs uma linguagem chamada DSL/Alpha para manipulação dos dados em tabelas relacionais. Pouco tempo depois da publicação de Codd, a IBM financiou um grupo para desenvolver um protótipo baseado nos conceitos e ideais elaborados por Codd. Este grupo criou uma versão simplificada do DSL/Alpha, que foi chamada de SQUARE. Refinamentos no SQUARE levaram à linguagem chamada SEQUEL que, finalmente, foi renomeada para SQL (BEAULIEU, 2009).

O SQL anda de mãos dadas com o modelo relacional (BEAULIEU, 2009), provendo recursos para a manipulação, gerenciamento e acesso aos dados contidos em um sistema de gerenciamento de banco de dados (SGBD).

Segundo Batista et al. (2019), para que a SQL consiga fornecer tantos recursos, ela precisa ser formada por várias partes, cada uma com seu propósito específico, tais são:

**DQL - Linguagem de Consulta de Dados:** Define os comandos responsáveis por recuperar os dados das tabelas do banco de dados, mais especificamente o comando SELECT e suas funções;

**DML - Linguagem de Manipulação de Dados:** Define os comandos responsáveis pela manipulação dos dados no banco (INSERT, UPDATE e DELETE);

**DDL - Linguagem de Definição de Dados:** Define os comandos responsáveis pela criação (CREATE) de tabelas, *views*, índices, atualização dessas estruturas (ALTER), assim como a remoção (DROP);

**DCL - Linguagem de Controle de Dados:** Define os comandos responsáveis por controlar os acesso aos dados do banco, adicionando (GRANT) e removendo (REVOKE) permissões de acesso;

**DTL - Linguagem de Transação de Dados:** são os comandos utilizados para gerenciar as transações executadas no banco de dados, como iniciar (BEGIN) uma nova transação, confirmar (COMMIT) as alterações feitas nela ou desfazer (ROLLBACK) tais alterações.

O foco do jogo aqui proposto é abordar a parte de DQL para consulta e visualização dos registros, querendo incentivar o estudante a dominar estratégias mais complexas de consultas em bancos de dados.

### 3.2 QUAL A IMPORTÂNCIA DA SQL

Quando falamos de bancos de dados, temos diversos bancos que são usados ao redor do mundo, Chand (2022) exibiu uma pesquisa que lista os bancos de dados mais usados do mundo, ranqueados entre 0 e 1400 usando os seguintes critérios:

- Número de resultados nas ferramentas de busca Google, Bing e Yandex.
- Frequência de buscas no *Google Trends*.
- Frequência de discussões técnicas nos tão bem conhecidos sites de TI como “Stack Overflow” e “DBA Stack Exchange”.
- Número de ofertas de emprego no “Indeed” e “Simply Hired”.
- Número de perfis profissionais em redes sociais como “Linkedin” e “Upwork”.
- Mais mencionados no Twitter.

Tabela 1 – Bancos de dados mais populares do mundo e seus rankings

Posição	Banco de Dados	Ranque
1	Oracle	1268.84
2	MySQL	1154.27
3	Microsoft SQL Server	1040.26
4	PostgreSQL	466.11
5	MongoDB	387.18
6	IBM Db2	179.85
7	Redis	149.01
8	Elasticsearch	143.44
9	Microsoft Access	141.62
10	SQLite	126.8

Fonte: Chand (2022)

A tabela 1 mostra uma lista dos 10 bancos de dados mais populares do mundo e seus respectivos ranques. É notável a discrepância do ranque do terceiro banco de dados mais usado para o quarto, com uma diferença de quase 600 pontos.

Dessa forma, podemos considerar para a análise de relevância da linguagem SQL para o mercado de tecnologia olhando para os 3 primeiros bancos de dados mais usados: Oracle, MySQL e Microsoft SQL Server.

Os três bancos de dados são relacionais e tem o SQL como sua linguagem de manipulação do banco (MYSQL, 2022) (ORACLE, 2015) (MICROSOFT, 2022), o que mostra como a linguagem é a mais usada no quesito banco de dados.

Além disso, SQL é usado por quase todos os grandes nomes da tecnologia, tal como Uber, Netflix e Airbnb (GRUPMAN, 2021). Mesmo entre as empresas como Facebook, Google e Amazon que construíram seus próprios sistemas de bancos de dados de alta performance, os times de análises de dados usam a SQL para buscar informações e realizar análises (GRUPMAN, 2021).

Se ainda não temos motivos suficientes para incentivar alguém da área de tecnologia a aprender SQL, podemos olhar para como as vagas de emprego exigem habilidade em SQL em sua descrição. A análise feita por (GRUPMAN, 2021) mostra que mais de 40% das vagas de emprego que envolvem dados do Indeed mencionam SQL, seguido de Python com pouco mais de 35%.

### 3.3 COMO SQL É ENSINADO

Existem vários livros e ferramentas que buscam orientar os estudantes em seu aprendizado da linguagem. Beaulieu (2009), por exemplo, segue o fluxo didático que pode ser visto na figura 4, primeiramente contextualizando o estudante sobre banco de dados relacional e a linguagem SQL, seguindo então para o ensinamento de como criar e popular um banco de dados (DML), chegando então a abordar os conteúdos referentes a DQL.

Figura 4 – Fluxo didático do Learning SQL



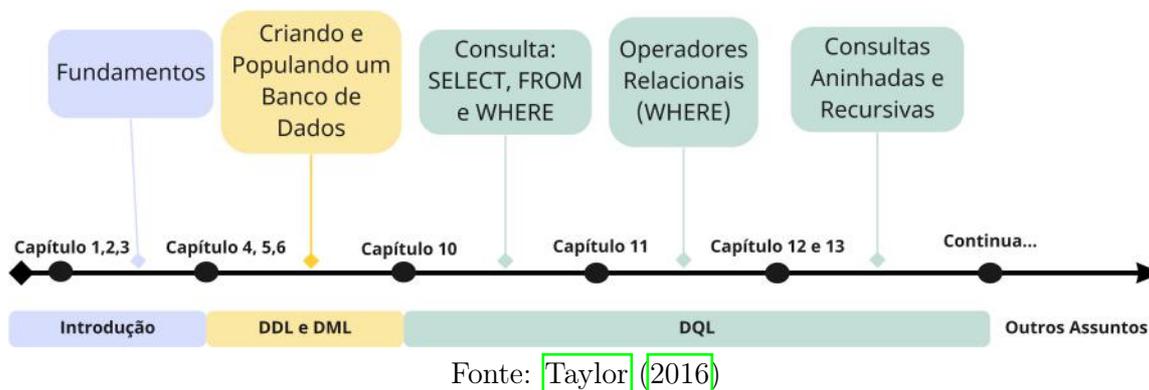
Fonte: Beaulieu (2009)

O autor então destrincha o conteúdo da DQL em alguns capítulos, começando com uma introdução a consultas, que ensina os comandos *select*, *from* e *where*, então explorando as diferentes regras que podem ser estabelecidas na cláusula *where*. Em seguida, o autor aborda a busca em múltiplas tabelas, introduzindo o comando *join*. Posteriormente ele aborda operações sobre conjuntos, tipos de dados, as funções de agrupamento e agregação, e, por fim, *subqueries*.

Taylor (2016) em seu livro “SQL para Leigos” também inicia o seu ensino com os fundamentos básicos, seguindo para os conteúdos de DML e DDL, ensinando o aluno a criar e popular um banco de dados e então seguindo uma sequência de assuntos bem semelhante a estruturada por Beaulieu (2009).

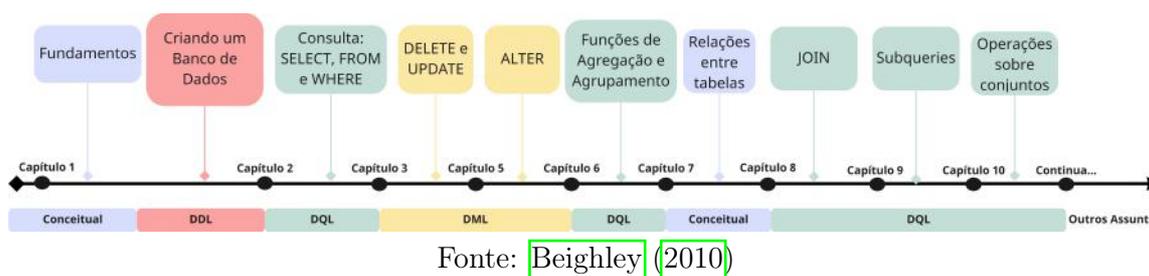
A figura 5 ilustra a sequência de conteúdos abordados por Taylor (2016) em seu livro “SQL para Leigos”.

Figura 5 – Fluxo didático do SQL para Leigos



Já Beighley (2010), em seu livro “Use a Cabeça! SQL”, faz uma abordagem bem diferente das duas acima, sendo baseado em pesquisas recentes de neurobiologia, ciência cognitiva e teoria do aprendizado. O fluxo didático geral do livro pode ser visto na figura 6.

Figura 6 – Fluxo didático do Use a Cabeça! SQL



Apesar de intercalar conteúdos de DML, DQL e DDL, Beighley (2010) também segue uma ordem de ensinamento semelhante aos dois anteriores, mas com uma diferença crucial: diferente do que é feito por Beaulieu (2009) em seu livro “Learning SQL”, a autora de “Use a Cabeça! SQL” ensina funções de agregação e agrupamento antes de ensinar *JOIN*. Acharmos interessante a abordagem da autora pois primeiramente ela explora tudo que se pode extrair de uma única tabela para só então ensinar como envolver mais de uma tabela em uma única consulta.

### 3.4 JOGOS PARA ENSINAR SQL

Motivados por um projeto acadêmico, Batista et al. (2019), uma equipe de alunos do curso de Ciência da Computação do Instituto Federal do Triângulo Mineiro, propuseram o desenvolvimento do SQL Planet, um jogo educativo para ensino de SQL.

O SQL Planet é um jogo online, disponibilizado como uma aplicação web, que está em fase de desenvolvimento. O estilo do jogo é aventura em texto (*text-based adventure*

*game*, em inglês), também conhecido como ficção interativa. É um gênero clássico em que toda interação entre jogo e jogador ocorre por meio de palavras na tela.

Para o sistema de banco de dados do “SQL Planet”, [Batista et al. \(2019\)](#) indicaram o uso do SQLite, já que é um banco de dados baseado em arquivos, o que permitia que uma nova instância do banco fosse facilmente criada para cada execução do jogo. Isso também permite que o jogo ocorra paralelamente sem que os jogadores interfiram no banco um dos outros, cada um possuindo a sua própria instância do banco de dados.

Outra vantagem do uso do SQLite apontado por [Batista et al. \(2019\)](#), é a impossibilidade de que solicitações maliciosas e Injeções SQL destruam o jogo.

Por esses mesmos motivos, optamos pelo uso do SQLite no desenvolvimento do jogo aqui proposto, permitindo que cada jogador possuísse sua própria instância do banco de dados e ao mesmo tempo protegendo o jogo de usuários maliciosos.

O “SQL Planet”, em seu escopo planejado, foca no aprendizado das partes de DQL, para consulta e visualização dos registros, e DML para manipulação e criação de registros.

Não encontramos referência para o jogo pronto, apenas sua proposta de desenvolvimento.

Um outro jogo educativo de SQL que por sua vez já possui sua implementação disponível online, é o “SQL Murder Mystery”, desenvolvido por [Germuska \(2019\)](#), é uma ficção interativa onde o jogador precisa montar consultas SQL para encontrar as informações relevantes para solucionar o caso de assassinato de acontece na história do jogo.

Com o jogo, é possível aprender chaves primárias e estrangeiras, funções de agregação e consultas em mais de uma tabela (*join*) [\(GERMUSKA, 2019\)](#).

O jogo também utiliza o SQLite como base do banco de dados. A temática do jogo é bem semelhante a temática aqui proposta, de resolver mistérios/crimes explorando informações no banco de dados.

Um outro jogo para ensino de SQL é o LudoSQL, desenvolvido por [Santos e Ribeiro \(2016\)](#), que tem como propósito servir de ferramenta de apoio ao aprendizado nas disciplinas introdutórias ao SQL. Segundo os autores, o jogo proposto proporciona a possibilidade de fixar de maneira lúdica o aprendizado adquirido em sala de aula.

O LudoSQL, diferente dos jogos citados anteriormente, possui uma interface gráfica e conteúdo amigáveis focados na escrita de comandos SQL, com o objetivo de desafiar o aluno a aprimorar seus conhecimentos sobre a linguagem. A dinâmica do jogo oferece, a cada execução, 10 desafios sorteados aleatoriamente para cada um dos quais o jogador tem um tempo determinado para solucionar [\(SANTOS; RIBEIRO, 2016\)](#).

Apesar de não ser diretamente especificado pelos autores, sua descrição e mecânicas parecem se encaixar em uma ficção interativa, com a história remetendo ao mundo de fantasia e magia.

O quadro [4](#) compara alguns aspectos dos jogos citados com o jogo aqui desenvolvido.

Quadro 4 – Comparação Jogos Educativos SQL

Jogo	Estilo	Temática	Implementação disponível	Tópicos	História
SQL Planet	Ficção Interativa	Aventura	Não	DML, DQL	Aventura num mundo desconhecido
SQL Murder Mystery	Ficção Interativa	Mistério	Sim	DQL	Solucionar um caso de assassinato
LudoSQL	Ficção Interativa	Aventura	Sim	DQL, DML, DDL	Resolução de testes num mundo de magia medieval
Nosso Jogo	Ficção Interativa	Mistério	Sim	DQL	Solucionar casos policiais e progredir na história

Fonte: Santos e Ribeiro (2016), Batista et al. (2019), Germuska (2019)

### 3.5 CONCLUSÃO

Com as informações levantadas e descritas nesse capítulo, temos já informações suficientes para responder à quarta e à quinta questão de pesquisa, que são, respectivamente, “(Q4) Por que ensinar SQL?” e “(Q5) Quais conteúdos didáticos de SQL o jogo deve cobrir?”.

O motivo do porquê ensinar SQL, pode ser entendido ao analisar a sua relevância para o mercado profissional mundial e ver que os maiores bancos de dados do mundo se utilizam da linguagem.

Já os conteúdos didáticos ensinados pelo jogo variam de acordo com a complexidade do mesmo e do seu planejamento, um jogo mais simples como o “LudoSQL” conseguiu cobrir um conjunto maior de conteúdos, enquanto que um jogo mais complexo como o SQL Murder Mystery e o SQL Planet, precisaram se ater a um escopo mais limitado.

O jogo aqui proposto então irá cobrir o conteúdo DQL, seguindo o fluxo de ensino proposto por Beighley (2010), começando consultas básicas (*SELECT* com *WHERE*), operadores relacionais (TAYLOR, 2016), funções de agregação e agrupamento, múltiplas tabelas (*JOIN*) e concluindo com consultas aninhadas (*subqueries*).

Esses serão os principais tópicos didáticos abordados no jogo, mas ainda terão conteúdos complementares que, conforme o jogador for evoluindo no jogo, ele acabará por entender e compreender, tais são: estrutura do banco de dados, chaves primárias e tipos de dados (numérico, data, *string*, etc).

## 4 PROPOSTA CONCEITUAL DE UM JOGO EDUCACIONAL SOBRE SQL

Este capítulo expõe o processo de elaboração e desenvolvimento do jogo.

Para planejamento do jogo foi utilizado o Endo-GDC, o *Game Desing Canvas* para jogos educativos endógenos descrito em [2.3.1](#).

Usando a plataforma MIRO (miro.com), foi possível simular o *canvas* ilustrado na Figura [1](#) e preenchê-lo de maneira colaborativa.

Na Figura [7](#), é possível ver o Endo-GDC completamente preenchido, dando uma visão geral do planejamento inicial do jogo.

### 4.1 ENDO-GDC

#### 4.1.1 Problemas

Os problemas que o jogo busca solucionar são: a falta de interesse dos alunos no aprendizado de Banco de Dados e o fato da linguagem de SQL fugir do padrão de linguagens procedurais normalmente ensinados em faculdade, sendo um dos principais fatores de dificuldade encontrados por estudantes de computação no aprendizado da linguagem (GARNER; MARIANI, 2015).

#### 4.1.2 Jogador/Aluno

As características do público alvo que desejamos atingir com o jogo foram:

**Faixa etária:** entre 15 e 30 anos.

**Profissão:** Estudante ou profissionais da área de computação.

**Gênero:** Todos os gêneros.

**Nível escolar:** Ensino médio e ensino superior.

**Cenário de Uso:** Computador.

#### 4.1.3 Conteúdo Pedagógico

O conteúdo pedagógico que o jogo busca abranger e trabalhar nos jogadores, são:

- a) Estrutura do banco de dados.
- b) Tipos de dados.
- c) Cláusulas de Busca.
- d) Consultas em SQL.

- e) Regras de Sintaxe.

#### 4.1.4 Objetivos de Aprendizado

Esta seção é importante para conseguir determinar se o jogo atingiu o seu propósito educativo (TAUCEI 2019). Além disso, servirá como um guia para a construção do jogo.

Os objetivos determinados para este jogo foram:

- a) **Procedural:** APLICAR as cláusulas de busca USANDO diferentes tabelas numa mesma consulta.
- b) **Factual:** LEMBRAR as regras de sintaxe IDENTIFICANDO os erros na escrita.
- c) **Conceitual:** ANALISAR a estrutura do banco de dados SELECIONANDO quais informações são relevantes para a fase.
- d) **Procedural:** CRIAR consultas em SQL CONSTRUINDO a query usando as cláusulas necessárias.
- e) **Factual:** LEMBRAR os tipos de dados IDENTIFICANDO as formas corretas de construir a *query*.

#### 4.1.5 Feedbacks Educativos

Definimos o feedback “Erro na busca”, que tem por objetivo indicar um erro de sintaxe ou de estrutura, onde o sistema de consultas, ao receber uma *query* SQL do jogador, analisará a sintaxe elaborada em busca de erros. Uma vez identificado, será sinalizado ao jogador para que tome consciência do problema.

Definimos também um feedback “tela de pistas”, que mostra ao jogador quais pistas ele já encontrou e quão longe ele está da solução final do caso, além do feedback de “pista encontrada”, que indica ao jogador que sua consulta SQL retornou uma informação importante para resolução do caso.

Ao encontrar todas as pistas, o jogador pode ver a conclusão final do caso na Tela de Pistas (referente ao feedback de andamento do jogo), posteriormente recebendo uma mensagem de algum personagem o parabenizando pela solução do caso.

#### 4.1.6 Inspirações

Para alinhar de maneira orgânica o lado técnico do aprendizado de SQL com uma estrutura narrativa cativante, procuramos entender melhor jogos existentes no mercado que conseguiram atingir o público com mecânicas equivalentes às principais estruturas da linguagem SQL.

O estilo de jogo que melhor conversa com esta proposta é conhecido como Visual Novel — uma narrativa interativa que oferece formas sucintas de interação. Entre os lançamentos de mais destaque nos últimos anos, Her Story (STEAMSPY, 2021) e Orwell

(STEAMSPY, 2016) apresentam estruturas que podem ser adaptadas com alto índice de aproveitamento para o ensino de SQL.

Desenvolvido por Sam Barlow e publicado em 2015 (STEAMSPY, 2021), Her Story trata de um jogo investigativo onde o jogador interage diretamente com um banco de dados fictício de uma delegacia policial para desvendar nuances do testemunho de uma suspeita.

O principal método de interação é a reprodução e manipulação de vídeos, em que podem ser ouvidas palavras-chave que são usadas em um sistema simples de busca.

Mesmo lançado de maneira independente, o jogo foi aclamado pela crítica especializada e aplaudido pelo público geral (IMDB, 2021).

No ano de 2015, Her Story conquistou seis nomeações ao BAFTA, British Academy of Film and Television Arts, das quais venceu em Best Game Debut e Best Game Innovation (IMDB, 2021). Segundo o site de análise SteamSpy (STEAMSPY, 2021), o jogo atualmente vendeu entre 1 milhão e 2 milhões de unidades apenas na plataforma Steam, o que demonstra o apreço do público por suas mecânicas de busca.

Por sua vez, Orwell foi um game publicado originalmente em 2016 (STEAMSPY, 2016), fruto do trabalho da desenvolvedora Osmotic Studios. Na trama, o jogador é responsável por operar um sistema que supervisiona todos os dados de uma organização para identificar criminosos que podem prejudicar a imagem do governo.

Segundo a sua descrição da Steam (2016), a proposta do jogo é adaptar as ideias revolucionárias do escritor George Orwell, que temia o avanço tecnológico como forma de oprimir e controlar a população por governos fascistas.

As mecânicas principais do jogo permitem selecionar palavras-chave em jornais e outras fontes para construir uma ficha criminal do suspeito. Aliado com as mecânicas de Her Story, esta estrutura resulta em uma maneira empolgante de explorar as principais estruturas de SQL.

Orwell foi um dos nomeados ao The Game Awards 2016 na categoria Games for Impact (STUDIOS, 2018). Segundo o site SteamSpy (2016), o título também se aproxima da marca de 2 milhões de cópias vendidas na plataforma Steam.

#### 4.1.7 História

Parte crucial para a melhor absorção do conteúdo proposto é uma narrativa que incentive o jogador-estudante a avançar em seus estudos. Deste modo, foi planejado um enredo de ficção policial com nuances de ficção científica para uma maior coesão com o tema.

Resumidamente, o jogador assume o papel de um estudante universitário do curso de Ciência da Computação que entra em contato com uma inteligência artificial com habilidades investigativas. Esta IA pede ajuda do jogador para solucionar um crime.

Ao completar seu requisito, o jogador é convidado a auxiliar a polícia local na Divisão de Inteligência. Um policial experiente é apontado como seu mentor e, após solucionarem uma certa quantidade de casos juntos, ele desaparece.

O mentor é acusado de um crime e torna-se o principal suspeito, o jogador então precisa usar todos os seus conhecimentos adquiridos ao longo da aventura para desvendar uma conspiração e provar a inocência de seu mentor.

#### 4.1.8 Objetivos do Jogo

O jogador será confrontado com casos investigativos a todo instante e seus objetivos são identificar os culpados e seus respectivos motivos utilizando buscas em SQL para justificar seu raciocínio.

Dessa forma determinamos os objetivos do jogo como sendo "Encontrar o motivo e o culpado pelo crime" e "Buscar informações direcionadas no banco para encontrar o culpado".

#### 4.1.9 Estética

As estéticas descrevem as respostas emocionais que se deseja evocar no jogador e os elementos que dão aparência sensorial ao jogo (TAUCEI, 2019).

O enredo foi construído para que o jogador tenha a oportunidade de se sentir traído e enganado pela sua própria corporação, germinando um desejo de lutar para provar a inocência de seu amigo. A intenção é criar uma motivação pessoal para perseverar na história, conectando o jogador ao personagem.

Por consequência, procura-se despertar a ansiedade do jogador por resolver cada novo mistério e, por sua vez, aprender SQL. A ideia de trabalhar com a polícia em casos visa nutrir um senso de responsabilidade e mérito que, de certo modo, é inerente a manipulação de bancos de dados em grandes empresas.

Poder escrever suas próprias consultas visa cultivar a liberdade do jogador. Assim como a possibilidade de navegar nos mistérios em seu próprio ritmo, respeitando seu tempo de aprendizado, mas tentando sempre manter a ansiedade do jogador em querer resolver logo o mistério.

Por fim, a capacidade de concluir seus mistérios usando conhecimentos de SQL adquiridos exclusivamente dentro do ambiente digital interativo gera uma sensação de realização, de conquista, que apenas os jogos podem proporcionar.

Já no aspecto de design do jogo, ele foi pensado para lembrar o formato de um computador, a fim de trazer uma sensação imersiva e mais próxima do "real".

#### 4.1.10 Dinâmicas

A interação do jogador com o universo do jogo é pautada pelas seguintes dinâmicas fundamentais. Primeiramente, é preciso investigar um caso policial tendo em mãos apenas uma tela de banco de dados e um “bloco de notas”.

Para investigar e encontrar pistas do caso policial, o jogador precisa analisar a estrutura do banco de dados para identificar corretamente as informações relevantes para a fase por meio de consultas SQL.

Além disso, o jogador poderá construir relações de amizade entre os personagens no progresso do jogo, interagindo com os mesmos conforme as opções de mensagens que pode enviar.

Dessa forma, o jogador consegue atuar como Cristiano Pereira (Cris), o estudante de computação que acabou por se envolver em casos policiais e precisa ajudar a polícia a resolver os casos usando seus conhecimentos de banco de dados.

#### 4.1.11 Mecânicas

O jogo é construído a partir da implementação das seguintes mecânicas:

**Bloco de template:** Um bloco de template para auxílio na criação de query (SELECT, JOIN, etc);

**Interface das tabelas user-friendly:** Tabelas fáceis de ler e cercadas de interações intuitivas permitem que o jogador concentre seu intelecto no processo de aprendizado;

**Escrever e rodar consultas SQL:** Ao que vamos chamar apenas de “consultar”, representa a escrita de comandos SQL;

**Board mental com as pistas selecionadas:** que chamaremos apenas de “Ver pistas”, transforma o progresso do jogador em um conceito visual, auxiliando a percepção de possíveis erros de pensamento;

**Apontar a solução do mistério:** Essencial para a conclusão dos objetivos. Testa as teorias do jogador e afere se o conteúdo narrativo e didático do segmento em questão foi compreendido;

**Interação via chat com os personagens da história:** Visa explorar a familiaridade do usuário com tecnologias de uso diário, aumentando a adesão de estudantes sem conhecimentos em videogames. Esta mecânica acopla duas mecânicas mais simples, sendo elas: “Ler Mensagem” e “Responder”.

#### 4.1.12 Restrições

Por fim, definimos as restrições para o desenvolvimento do jogo e tais foram:

**Unity e SQLite:** A engine que escolhemos para o desenvolvimento foi a Unity, por já termos experiência com ela como explicado no tópico [2.4](#). Já o SQLite foi utilizado por ser um banco de dados baseado em arquivo e pela impossibilidade de solicitações maliciosas como explicado no tópico [3.4](#).

**Computador:** O jogo será desenvolvido para computador, cobrindo o cenário de uso especificado na parte de "Jogador/Aluno"

**Jogo Digital:** O jogo será digital.

**Equipe não multidisciplinar:** Diferente de como normalmente funciona com desenvolvimento de jogos sérios ([TAUCEI, 2019](#)), não teremos uma equipe multidisciplinar, tendo apenas dois desenvolvedores para fazer o *design*, definir a interface e estética, implementar, testar e avaliar o jogo como um todo.

**TCC:** O TCC acaba sendo uma restrição impondo um prazo e um formato para o jogo, mas sendo também seu principal motivador.

## 4.2 6-11 FRAMEWORK

### 4.2.1 Emoções e instintos

Conforme visto no tópico [4.1.9](#) conseguimos identificar as seguintes emoções e instintos:

- Identificação - ao interagir com os personagens como o personagem principal, o jogador se identifica com ele;
- Raiva - gerada pela traição que acontece na história do jogo;
- Agressividade - consequência da traição;
- Vingança - consequência da traição;
- Curiosidade - para saber o desfecho dos casos policiais;
- Coleta - ao buscar e encontrar pistas;
- Ganância - ao querer encontrar mais e mais pistas;
- Orgulho - ao conseguir resolver os casos e coletar pistas;
- Excitação - resultado da curiosidade e do desejo de vingança pela traição;
- Proteção/Cuidado - ao tentar inocentar seu amigo acusado injustamente;
- Tristeza - por seu amigo ser acusado;

- Comunicação - conversar e interagir com personagens.
- Alegria - ao ser elogiado e interagir com personagens.

#### 4.2.2 Dinâmicas

Conforme visto em detalhes no tópico [4.1.10](#), temos as seguintes dinâmicas identificadas para o 6-11 *Framework* (DILLON, 2011):

**Investigar** - usa a mecânica “Consultar”;

**Analisar** - usa a mecânica “Ver pistas” para entender o que precisa investigar;

**Conversar** - usa as mecânicas “Ler Mensagem” e “Responder”;

**Resolver Caso** - usa a mecânica “Apontar a solução do mistério”.

#### 4.2.3 Diagrama

Levando em conta as estéticas e dinâmicas levantadas, temos na figura [8](#) o 6-11 *Framework*.

### 4.3 CONCLUSÃO

Conforme pôde ser visto no decorrer desse capítulo, o Endo-GDC nos provê informações suficientes para definir no detalhe como se dará o jogo, definindo suas inspirações, história, objetivos, estética, dinâmicas e mecânicas, todos pontos fundamentais para a elaboração de um jogo, além de nos proporcionar uma visão detalhada de todas as restrições envolvidas no processo de desenvolvimento do mesmo.

Já no aspecto educacional, conseguimos determinar o perfil de jogador/aluno que gostaríamos de atingir, o problema que o jogo visa solucionar no processo de aprendizado, o conteúdo pedagógico que deve ser abordado pelo mesmo e, por fim, os objetivos de aprendizado que devem ser promovidos pelo jogo.

Com todas essas informações especificadas, é possível começar o desenvolvimento do jogo, sempre se atentando em seguir os pontos determinados no Endo-GDC.

Figura 7 – Modelo ENDO-GDC Preenchido

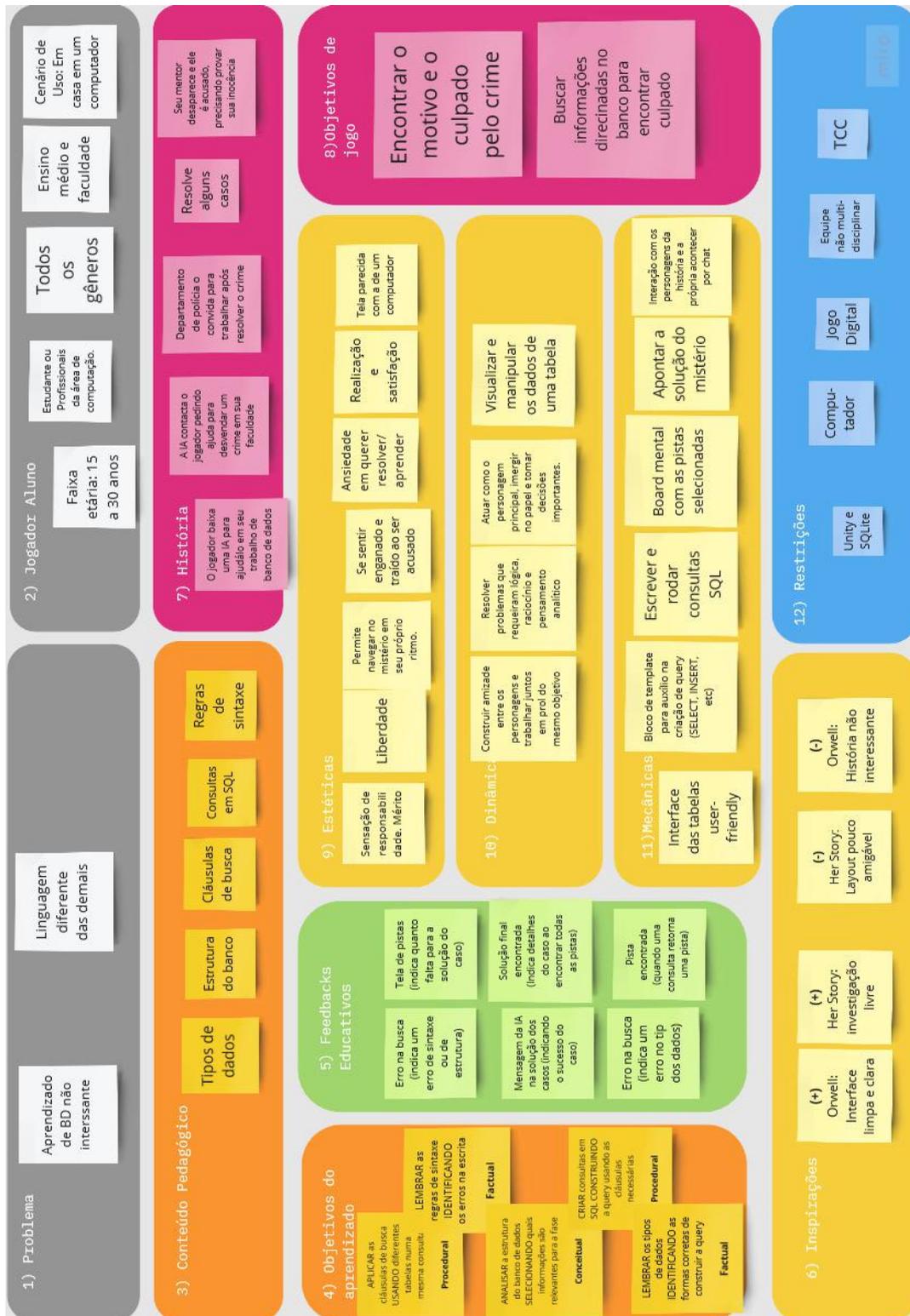
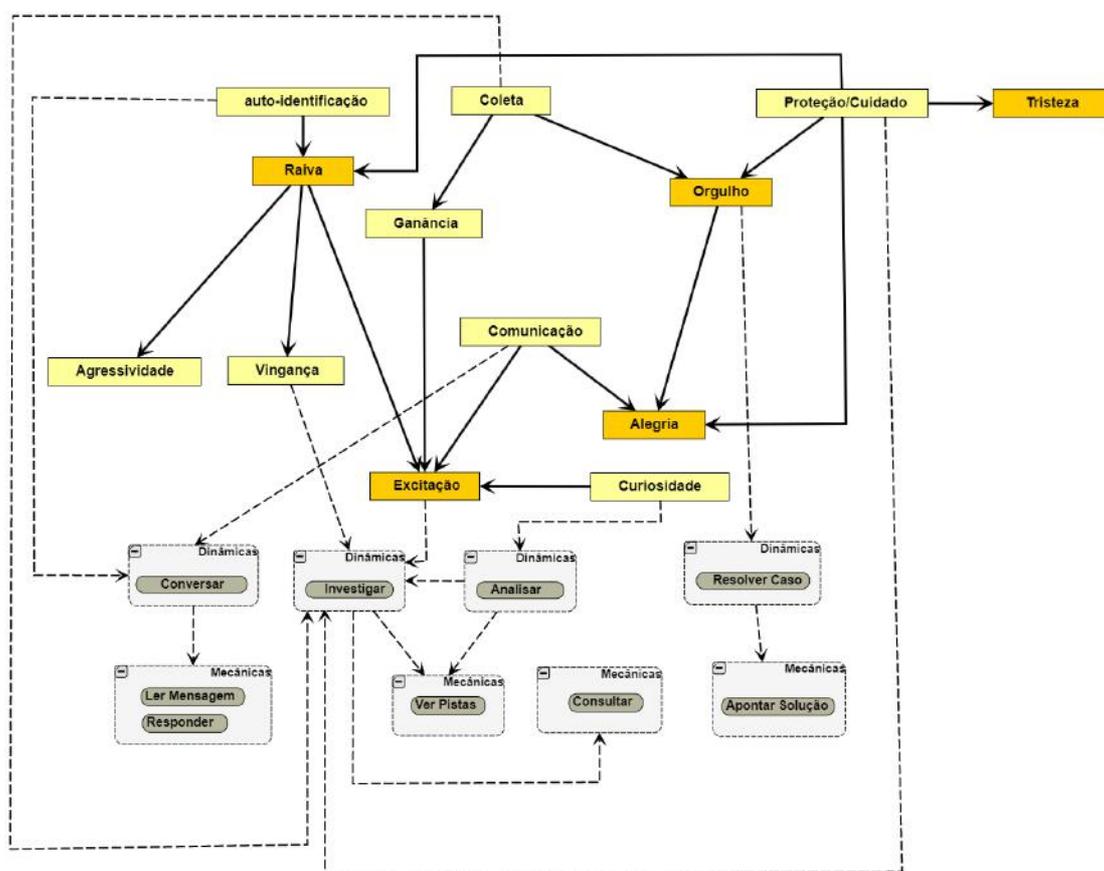


Figura 8 – 6-11 Framework



## 5 IMPLEMENTAÇÃO DE UM MVP DO SQL-GAME

### 5.1 HISTÓRIA

#### 5.1.1 A jornada do herói

Campbell (1949) em seu livro “The Hero with a Thousand Faces”, discute a jornada do herói arquetípico encontrada em inúmeras mitologias e culturas do mundo todo. Ele então explora a teoria de que tais mitos compartilham de uma estrutura fundamental, a qual Campbell chamou de “monomito”, descrita da seguinte forma:

Um herói se arrisca a sair do seu dia-a-dia comum para uma região de maravilha sobrenatural: forças fabulosas estão lá para serem encontradas e uma vitória decisiva está a ser ganhada: o herói volta a partir desta misteriosa aventura com o poder de conceder bênçãos sobre seus companheiros.

Campbell (1949) então definiu 17 etapas que compunham este monomito, as quais eram divididas em 3 grandes áreas:

- **A partida**
  1. O chamado à aventura
  2. A recusa do chamado
  3. Ajuda sobrenatural
  4. A travessia do limiar
  5. O ventre da baleia
- **A iniciação**
  6. O caminho das provas
  7. O encontro com a deusa
  8. A mulher como tentação
  9. Expição com o Pai
  10. A apoteose
  11. A última dádiva
- **O retorno**
  12. A recusa do retorno
  13. O voo mágico

14. Resgate interior
15. A travessia do limiar de retorno
16. Mestre de dois mundos
17. Liberdade para viver

Já em 1992, [Vogler \(2007\)](#) condensou a jornada de 17 etapas definida por Joseph John Campbell, em 12 etapas em seu livro “A Jornada do Escritor”, de forma a conduzir ao sucesso na construção de uma história.

As jornadas definidas por [\(VOGLER, 2007\)](#) são:

**O mundo comum** é o ambiente inicial, mostra quem é o personagem, onde ele vive e com quem se relaciona. É onde o personagem é apresentado de maneira geral.

**O chamado à aventura** é o momento da história onde o protagonista tem um desafio que o obriga a experimentar coisas novas.

**Recusa do chamado** o herói se vê em conflito entre sair da sua zona de conforto e os desafios que enfrentará. Acaba por recusar o chamado, mas continua incomodado.

**Encontro com o mentor** , diante do impasse, o encontro com o mentor lhe dá o empurrão necessário para embarcar na “aventura”.

**Travessia do primeiro limiar** é o momento em que ele entra na aventura e parte em rumo ao desconhecido, podendo ser um lugar de fato, uma nova habilidade, entre outros.

**Provas, aliados e inimigos** agora que o herói está em sua estrada, ele encontra obstáculos, contratempos e desafios menores que testam suas habilidades, além disso, a visão sobre o personagem é aprofundada e ele adquire a capacidade de discernir entre aliados e inimigos.

**Aproximação da caverna secreta** o herói faz uma espécie de pausa, se abrigando em seu esconderijo (interior ou não), retornando a seus questionamentos iniciais e enfrentamento dos medos.

**A provação** é uma espécie de “morte” a qual o herói precisará passar para alcançar seus objetivos, tem um significado de transformação e o herói precisará usar todos os conhecimentos e experiências adquiridos.

**A recompensa** simboliza sua transformação em uma pessoa mais forte e pode ser representada por um objeto de grande valor, a reconciliação com alguém, um novo conhecimento, etc.

**O caminho de volta** não oferece tantos perigos mas sim um momento de reflexão onde o herói se vê diante de uma escolha entre a realização de um objetivo pessoal ou um bem coletivo maior.

**A ressurreição** é o ponto mais alto da história, em que o inimigo ressurge quando ninguém mais esperava, em um ataque que ameaça mais do que ele, afetando todos a sua volta.

**O retorno com o elixir** representa o momento de reconhecimento efetivo do herói. Simboliza seu sucesso, conquista e mudança e deixa muito claro para todos que as coisas nunca mais serão as mesmas por ali.

### 5.1.2 A história do jogo

A história do jogo foi elaborada usando a Jornada do Herói de [Vogler \(2007\)](#), seguindo as 12 etapas por ele especificadas e descritas no tópico [5.1.1](#)

Abaixo estão especificados os eventos do jogo que acontecem em cada etapa da jornada.

**O mundo comum** Na história do jogo definimos o personagem principal como sendo Cris, estudante de computação numa faculdade renomada. Entrou há pouco mais de um ano e tem conseguido se virar em todas as matérias. Tem uma autoestima baixa e isso faz com que questione suas habilidades com frequência.

O mundo comum dele é apresentado quando seu amigo entra em contato e pede ajuda para fazer um trabalho da faculdade sobre banco de dados.

**O chamado à aventura** Então uma IA misteriosa e simpática lhe manda mensagem, lhe pedindo ajuda. Ela fala que aconteceu um crime no seu campus, mas ninguém consegue encontrar o culpado. Quando Cris resolve o caso, acaba parando em todos os canais regionais de notícia e “viraliza” na internet como gênio investigador. O Secretário da Polícia Civil, Sr. Patrocínio lhe convida para a polícia, sendo este o seu chamado à aventura.

**Recusa do chamado** O cargo é para perito criminal em computadores. A polícia precisa de um novo investigador criminal especializado em SQL, e Cris parece ser o candidato perfeito. Cris vai precisar aprender SQL para isso e não sabe se vai conseguir. Cris diz que não pode aceitar o cargo, mas o investigador que vai ser seu parceiro aparece e pede ajuda com o caso que ele está trabalhando.

**Encontro com o mentor** O investigador é conhecido como Soares, o melhor da área. Boa parte das evidências no banco de dados foi ele quem coletou, então ele consegue dizer mais ou menos onde procurar, tranquilizando Cris quanto à sua responsabilidade.

**Travessia do primeiro limiar** Após a conclusão do primeiro caso, Soares lhe entrega um novo e diz que precisa atender o chamado para uma outra investigação de homicídio, mas continuará a lhe ajudar por mensagens. Cris tem acesso aos recursos da polícia, com acesso parcial ao banco de dados. Seguindo as pistas fornecidas por Soares, Cris acaba resolvendo seu primeiro caso oficial, o que impressiona seus superiores.

**Provas, aliados e inimigos** Soares lhe manda uma mensagem com seu próximo caso, junto com um pedido de desculpas por estar tão ocupado. Após isso, ele não responde mais suas mensagens. O caso é o homicídio que ele saiu para investigar anteriormente.

O caso ocorreu em um shopping e todas as evidências apontam para um suspeito, mas tem algo de estranho nesse caso. Pelo menos uma prova o inocenta das acusações. O delegado (sr. Patrocínio), parabeniza Cris pelo trabalho e quando é levantada a inconsistência, ele diz ser normal. Ele lembra que a função de Cris como perito é desenterrar provas, e que o juiz é que iria dizer se o suspeito é culpado ou não.

**Aproximação da caverna secreta** Ainda sem nenhum sinal de Soares, Cris recebe seu novo caso direto do delegado, o Sr. Patrocínio, o qual se trata de um pedido da Delegacia de Descoberta de Parapeiros. Cris está encarregado de levantar informações sobre o desaparecimento de uma deputada conhecida por enfrentar milicianos e percebe que algumas informações estavam censuradas. Alguém tentava encobrir os casos de milícia reportados.

Depois de muito procurar, Cris acha que descobriu onde ela possa estar e uma equipe policial é mobilizada. Lá, eles encontram a deputada morta e Soares na cena do crime. Ele jura ser inocente e que estão o incriminando. Soares é afastado enquanto instauram um inquérito por assassinato.

**A provação** Todos lhe dizem como Soares é corrupto e um péssimo policial, mas alguma coisa diz a Cris para confiar em seu mentor. Algo estava muito errado e o banco de dados mostrava isso. O delegado pede para que Cris trabalhe em um novo caso e que não se preocupe com Soares, pois a justiça seria feita.

Cris então decide insistir e investigar todo o banco de dados das delegacias, ao encontrar alguma das pistas relevantes sobre o suspeito, o jogo será interrompido com uma mensagem do Soares.

**A recompensa** Soares diz que sabe que está tentando inocentá-lo e agradece pela ajuda de Cris, e fala que é inocente e que somente Cris pode ajudá-lo. Logo após, recebe uma mensagem do Sr. Patrocínio.

**O caminho de volta** A mensagem do Sr. Patrocínio intimida Cris a escolher ajudar Soares ou continuar no emprego. Cris escolhe ajudar Soares.

**A ressurreição** Ao escolher ajudar Soares, o Sr. patrocínio demite Cris. Porém ao ligar seu computador de Casa, Lexa ainda está com ele, e informa que tomou a liberdade de fazer o backup do banco de dados da polícia, permitindo que ele continuasse com sua investigação. Cris resolve o caso inocentando Soares, e descobre que Sr. Patrocínio como mandante dos policiais corruptos.

**O retorno com o elixir** Ainda em casa, Cris destranca a matrícula na faculdade para continuar seus estudos. Dessa vez ele vai começar a matéria de Banco de Dados II. Lexa aparece para se despedir. Cris fica muito abalado e afirma que ele precisa dela. Lexa diz que ele já aprendeu tudo que precisava para conseguir se virar sozinho. Cris já provou a si mesmo que é capaz.

## 5.2 PROTOTIPAGEM

Como especificado na seção [4.1.9](#), a interface do jogo com o usuário tinha como objetivo simular um computador, dessa forma definimos as seguintes telas, prototipando-as usando o Figma, uma ferramenta de design colaborativo.

**A Tela principal** é a tela principal do jogo, onde o jogador consegue acessar todas as outras telas ou mesmo voltar ao menu.

Colocamos como imagem de fundo a imagem padrão do Windows XP, colocando botões para abrir cada das demais telas do jogo. A Tela Principal pode ser vista na figura [9](#)

**A Tela de Conversas** é por onde o jogador interage com a história do jogo e com os demais personagens da mesma.

Essa tela foi elaborada no formato de um celular, para se assemelhar a um chat de conversas, sendo internamente dividida em duas telas. A primeira, conforme pode ser vista na figura [10](#), é a tela de contatos, onde o jogador pode ver todos os personagens que já lhe enviaram mensagem e selecionar conversas para visualizar ou interagir.

A segunda tela dessa janela é a Tela de Mensagens, que pode ser vista na figura [11](#), com mensagens do jogador e do personagem distinguidas pela cor.

Na parte inferior dessa tela aparecem as opções de escolha de resposta do jogador e na parte superior uma identificação do personagem.

**Tela de Pistas** é onde o jogador obtém um *feedback* do seu desenvolvimento na fase e quanto ele ainda precisa atingir.

Figura 9 – Tela Principal

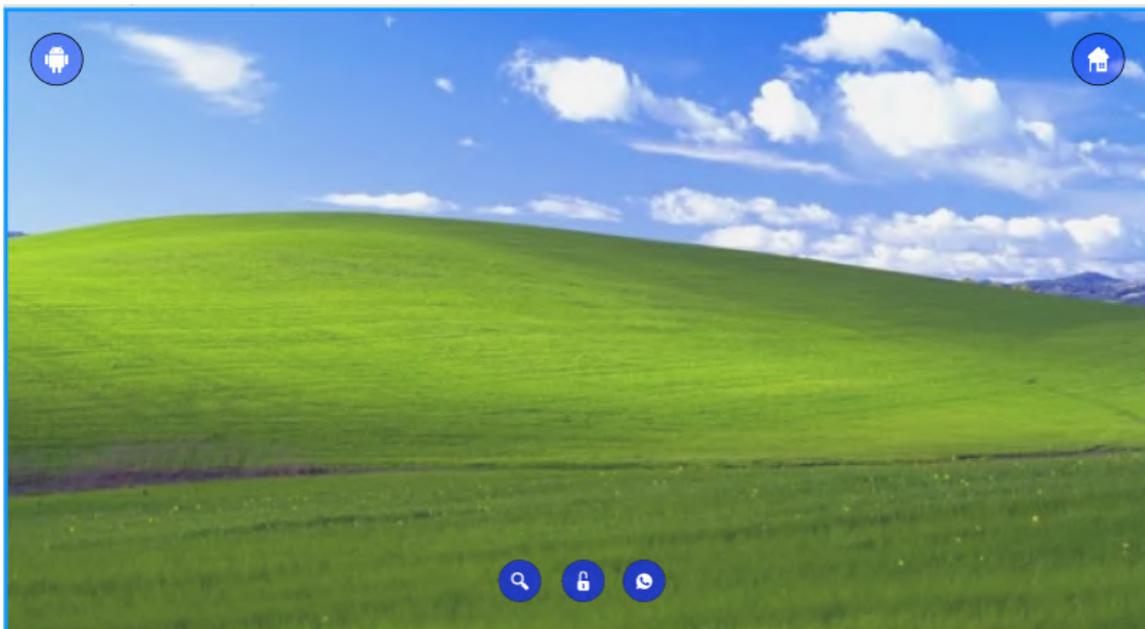
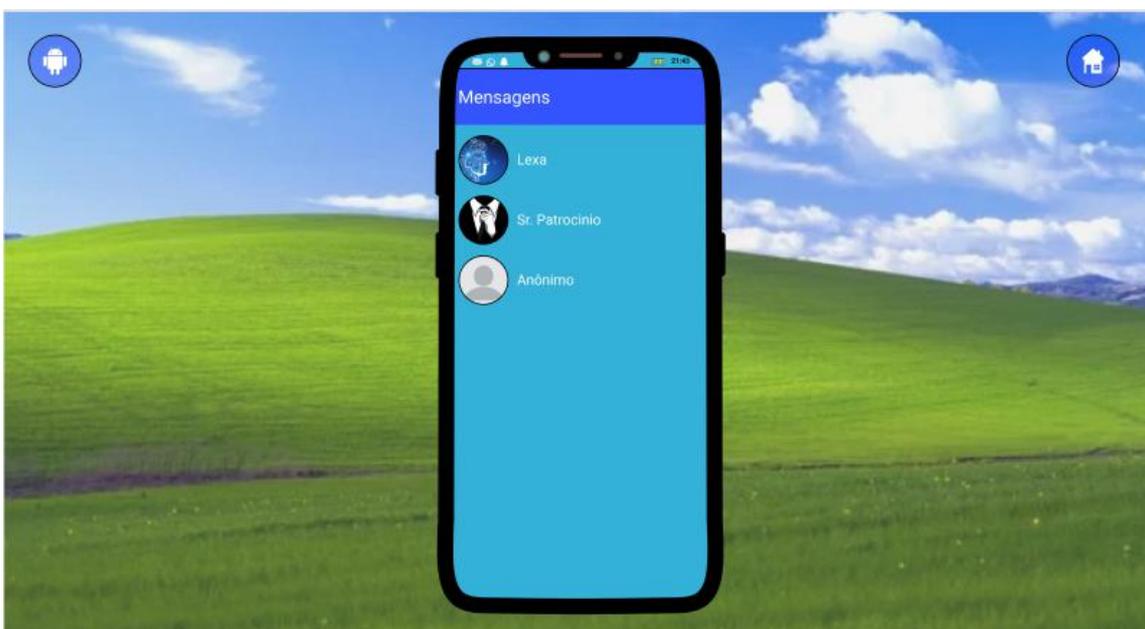


Figura 10 – Tela de Contatos



A tela simula um "bloco de notas" com anotações de "dicas" para a resolução do mistério suas conclusões. A tela de pistas pode ser vista na figura [12](#).

**Tela de Tabelas** É a tela onde são realizadas as consultas SQL, sendo a principal tela do jogo responsável pelo progresso do jogador.

A tela mostrada na figura [13](#) possui 4 funcionalidades básicas: um campo de busca para escrita do comando SQL, um combo box que permite selecionar um modelo

Figura 11 – Tela de Mensagens

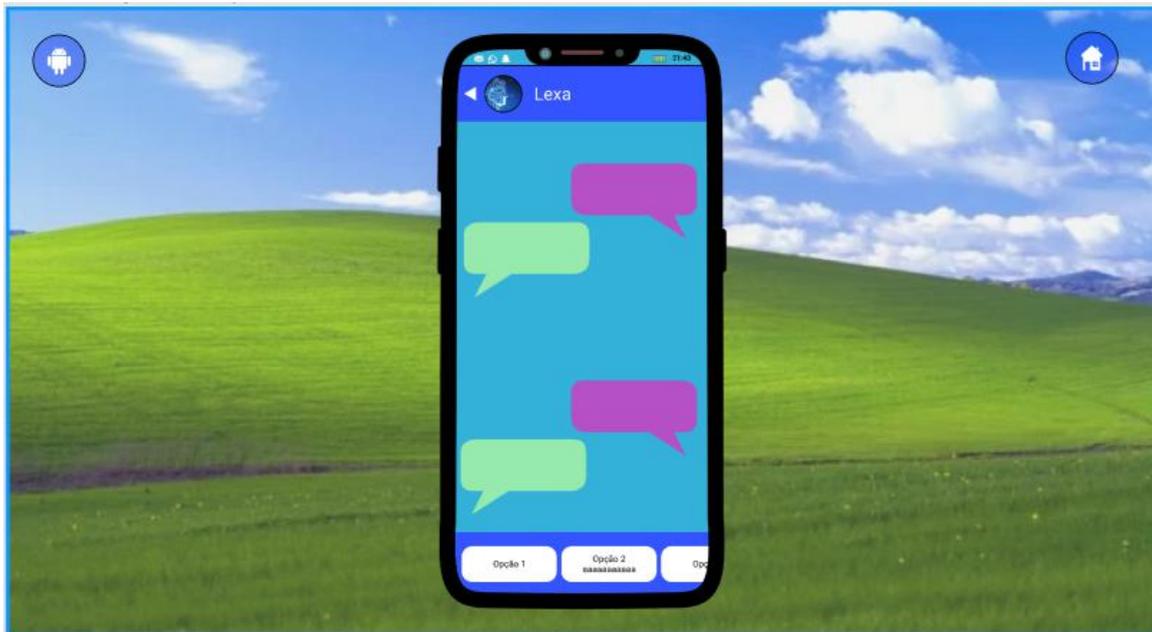
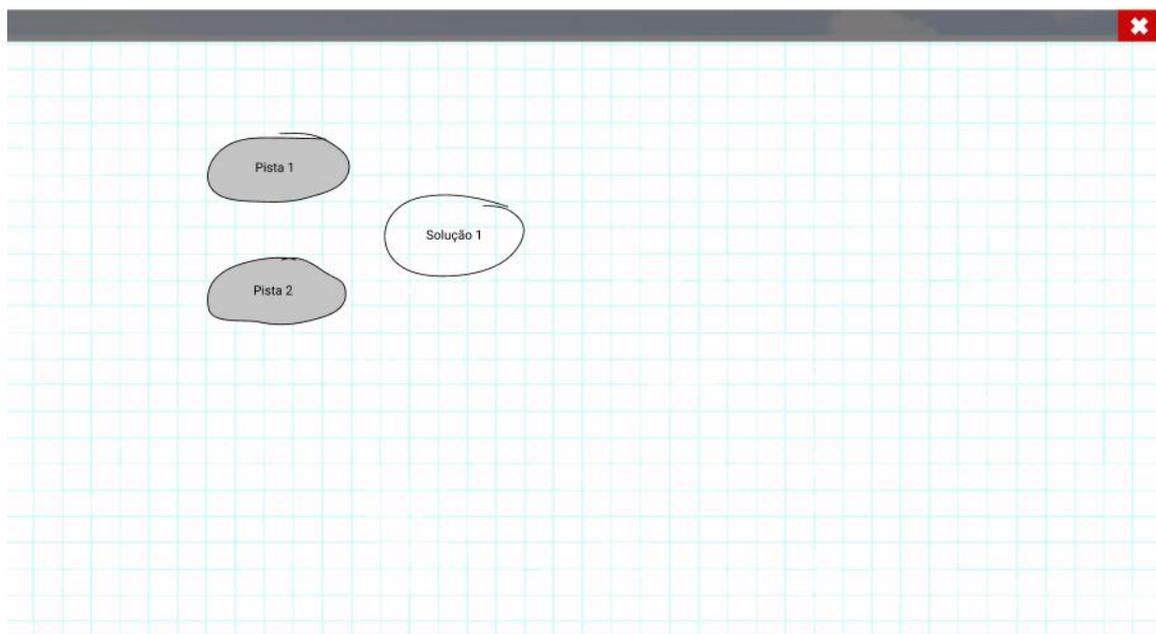


Figura 12 – Tela de Pistas

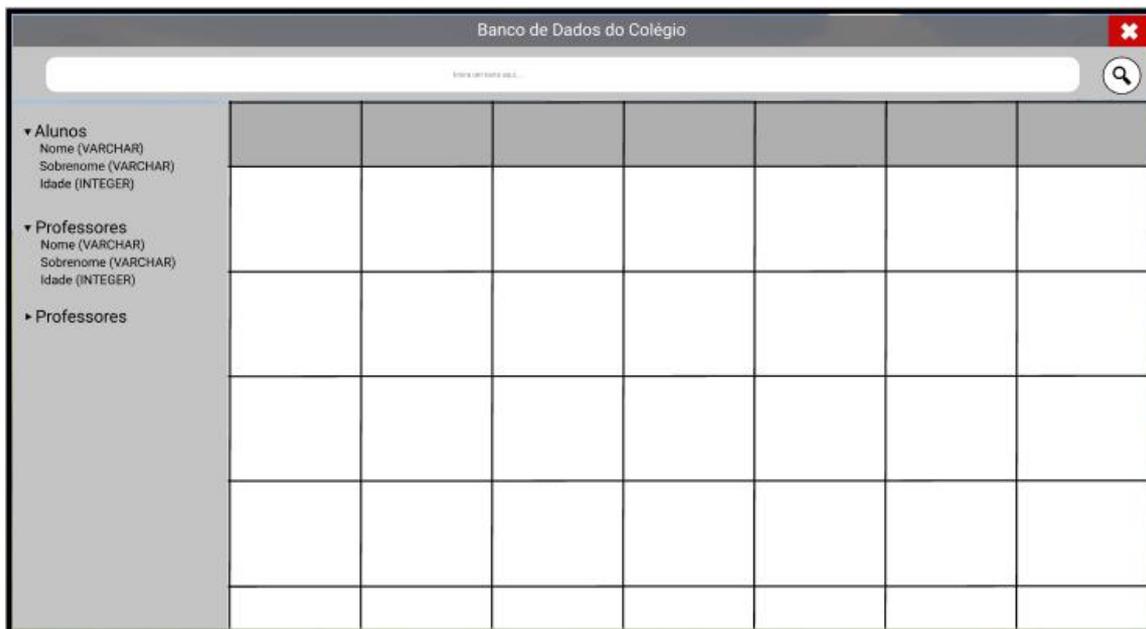


de pesquisa, uma área de dados do banco, que lista as tabelas e suas colunas e, por fim, uma área que exibe os dados de retorno da pesquisa em formato de tabela.

### 5.3 ESQUEMA DE FASES

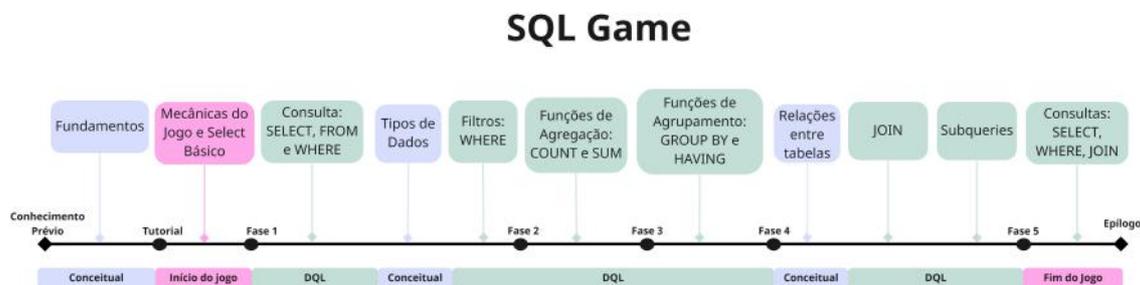
Como citado no Capítulo 3 existem varias formas de se ensinar SQL, o jogo aqui proposto irá cobrir o conteúdo DQL, começando com consultas básicas (*SELECT* com

Figura 13 – Tela de Tabelas



*WHERE*), operadores relacionais (TAYLOR 2016), funções de agregação e agrupamento, múltiplas tabelas (*JOIN*) e concluindo com consultas aninhadas (*subqueries*).

Figura 14 – Fluxo Didático SQLGame



Na figura 14 é possível ver de maneira geral o fluxo didático coberto por cada uma das fases.

Inicialmente consideramos que o jogador deverá ter conhecimento sobre os fundamentos de bancos de dados (cobrindo as etapas de Fundamentos/Background de Beighley (2010), Taylor (2016) e Beaulieu (2009)) pois o jogo se limita a ensinar os conceitos de DQL. Todo o conhecimento adicional que o jogador necessitar e ainda não souber, o mesmo terá que buscar uma forma de aprender (seja na internet ou em um livro) de forma que o conteúdo didático necessário para o cumprimento do jogo não estará incluso no *Minimal Viable Product* (MVP).

O jogo foi composto de 5 fases mais um tutorial, este tutorial tem como objetivo ensinar a mecânica do jogo, e como o mesmo deve ser jogado, apenas exigindo do jogador

aprender a fazer uma consulta SQL simples (*SELECT \* FROM table WHERE filtro*), tudo isso é passado na história como sendo um trabalho da faculdade do jogador.

Já na fase 1, uma inteligência artificial entra em contato com o jogador, pedindo sua ajuda para resolver um crime que havia acontecido em sua faculdade, o mesmo concorda em ajudar e precisar descobrir o culpado por um roubo de computadores. Aqui os objetivos de aprendizado começam a ser aplicados, começando pelo segundo ponto especificado por Taylor (2016), os operadores relacionais, com isso o jogador começará a precisar fazer consultas com filtros mais específicos, precisando realizar filtragem por data, por *string*, ou com mais de um filtro ao mesmo tempo.

Além disso, nessa fase o jogador consegue, implicitamente, obter um entendimento dos diferentes tipos de dados (*VARCHAR*, *DATE*, *DATETIME* e *INTEGER*) que o banco possui e como eles afetam a escrita da consulta SQL desejada.

Na fase 2, após resolver o crime, ele é convidado para ser estagiário na delegacia de polícia local para trabalhar exclusivamente com banco de dados, e recebe seu primeiro caso oficial, um caso de corrupção em uma empresa, onde ele precisará descobrir quem estaria desviando o dinheiro da mesma. Agora que o jogador já sabe fazer as cláusulas *WHERE* com clareza, o mesmo será desafiado a utilizar funções de agregação como *SUM* e *COUNT* além de ser necessário filtrar as colunas do banco pois o mesmo tem muitas colunas (uma restrição propositalmente adicionada), o forçando a melhorar seus conhecimentos na cláusula *SELECT*.

Na fase 3 o jogador se deparará com um caso de assassinato, sendo um caso um pouco mais complexo, e que o jogador pode se deparar com pistas que fazem parecer que o suspeito é inocente, deixando uma dúvida no ar. Nesta fase o jogador precisará aprender a usar a função de agrupamento *GROUP BY* e fazer uso da cláusula *HAVING*.

No início da fase 4, para surpresa do jogador, seu mentor Soares desaparece, mas o mesmo tem trabalho a fazer, precisando encontrar uma pessoa desaparecida, o que acaba por abordar um tema sensível relacionado a milícia. Nesta fase é necessário ao jogador aprender a utilizar a cláusula *JOIN* e consultas aninhadas, para pelo menos uma das consultas. Ao final desta fase, o jogador descobre que Soares está sendo acusado de ser o responsável pela morte da pessoa desaparecida, pois o mesmo estava na cena do crime.

Semelhante ao que acontece na fase 1, essa fase ajuda o jogador a entender um pouco mais sobre as relações entre tabelas e como juntá-las corretamente em uma única consulta.

Por fim, na fase 5 o jogador decide tentar ajudar seu mentor, que ele acredita ser inocente, e, com ajuda da Inteligência Artificial, remove as censuras do banco de dados da fase anterior e consegue acesso às mensagens da vítima. Esta fase funciona como um fechamento para a história e ele precisará utilizar diversos conhecimentos adquiridos ao longo do jogo para resolver o seu último caso.

Durante esta fase, ao encontrar a pista de que o seu chefe é um dos milicianos, o mesmo remove os acessos do jogador e o demite, mas com a ajuda da inteligência artificial, o

jogador pode terminar o caso e fechar a história.

Após isso o jogador recebe algumas mensagens para fechamento da história que funcionam como um epílogo.

#### 5.4 PROVA DE CONCEITO

Durante o desenvolvimento do jogo, foi criada uma prova de conceito (POC) para checarmos se o jogo iria atender o objetivo desejado e se seu desenvolvimento estava indo pelo o caminho certo. Nesta POC implementamos a fase 1 e o tutorial e pedimos para uma pessoa com conhecimentos de SQL testar o jogo como se o mesmo tivesse o nível de conhecimento previsto para as fases em questão.

Como resultado foi possível identificar alguns *bugs* e vários pontos de melhora na jogabilidade, sendo eles:

**Histórico de consultas** que permite que o jogador recupere pesquisas realizadas anteriormente através de dois botões de “anterior” e “próximo”.

**Menu de templates** para auxiliar no uso de diversas funções que o jogador não conhece, permitindo-o selecionar “modelos de consulta” e apenas personalizá-los de acordo com a necessidade. Esta funcionalidade já estava prevista no Endo-GDC do jogo, mas como não foi implementada na POC, acabou se mostrando um ponto relevante e do qual o jogador sentiu falta.

**Feedback de Mensagem** para informar ao jogador que ainda há mensagens a serem recebidas no chat em questão.

**Função Novo Jogo Sem Tutorial** essa função foi pedida justamente por a fase tutorial ser desnecessária para quem já tem algum conhecimento de SQL e está familiarizado com a mecânica.

#### 5.5 PRODUTO VIÁVEL MÍNIMO - MVP

Após as alterações da Prova de Conceito (5.4) e implementação das demais funcionalidades do jogo, fechamos a sua primeira versão completa para disponibilizar para teste.

Esta versão, após diversos ajustes e modificações, muitas das quais com o objetivo de se adequar às limitações da *engine* utilizada, cobriu a história completa especificada no tópico (5.1) e todas as fases elaboradas e descritas no tópico (5.3)

**Tela inicial** A tela inicial do jogo foi feita de maneira bem simples, permitindo apenas ao jogador continuar o jogo anteriormente salvo, criar um novo jogo, ou criar um

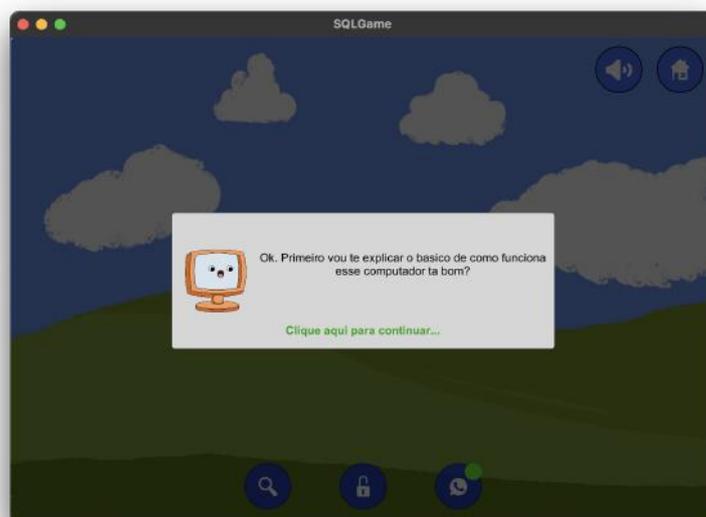
Figura 15 – Tela Inicial



novo jogo sem tutorial (funcionalidade pedida na POC para quem queria começar o jogo direto e já sabia o básico do mesmo).

**Tutorial** A funcionalidade de tutorial foi implementada como uma janela com texto instrutivo sobre o jogo, explicando como funciona a dinâmica de fases e como interagir com cada uma das telas.

Figura 16 – Tutorial



A funcionalidade de tutorial passa por cada etapa conforme o jogador clica em “con-

Figura 17 – Tutorial com Foco

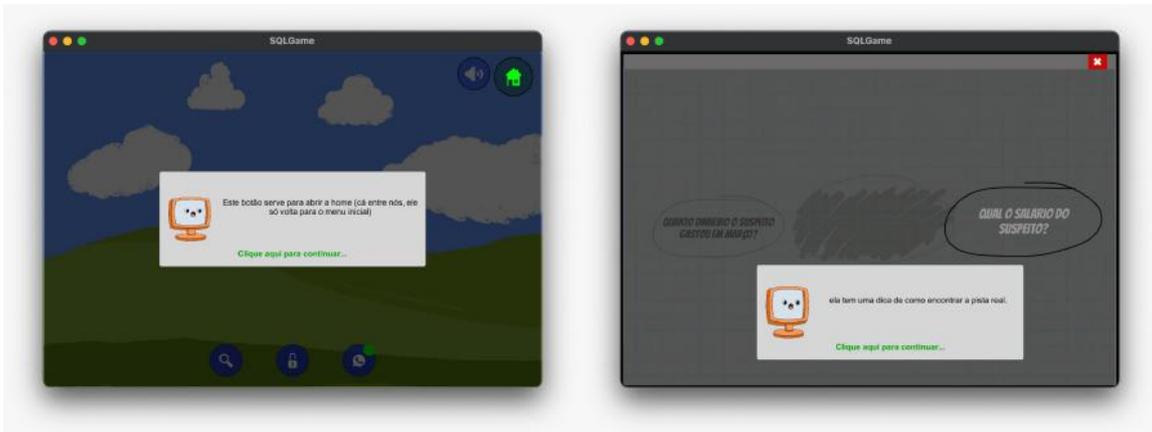


Figura 18 – Botão em foco

Figura 19 – Pista em foco

tinuar”, tendo uma funcionalidade de “*focus*”, que anima um determinado elemento de UI alterando seu tamanho e cor para indicar ao jogador ao que se refere a etapa atual do tutorial.

As figuras 18 e 19 mostram dois exemplos de tutorial com o uso do foco para indicar ao jogador ao que se refere a instrução dada. A figura 18 mostra uma instrução sobre um botão, que tem seu tamanho e cor aumentado e diminuindo em *loop*, chamando a atenção do jogador sobre o mesmo.

Já o exemplo ilustrado na figura 19, é o tutorial da tela de pistas, cujo objeto de foco em vez de um botão é o próprio elemento de pista na tela, explicando como ele funciona e o que representa.

Dessa forma as mecânicas do jogo conseguem ser explicadas adequadamente.

**Chat** A tela de Chat simula uma conversa pelo celular do jogador, conforme pode ser visto na figura 20. Esta funcionalidade dispõe de um feedback que indica se ainda há mensagens a serem recebidas pelo jogador na forma de um texto “Digitando...” no canto superior direito do celular.

Além disso, há uma imagem e nome do contato ainda no cabeçalho do celular, e um botão que permite voltar para a tela de contatos e, em seguida, sair do celular.

As mensagens do jogador e do NPC são diferenciadas pela cor e sempre há um *delay* de 3 segundos entre uma mensagem e outra do NPC, indicando o período que ele levaria para “digitar” a mensagem.

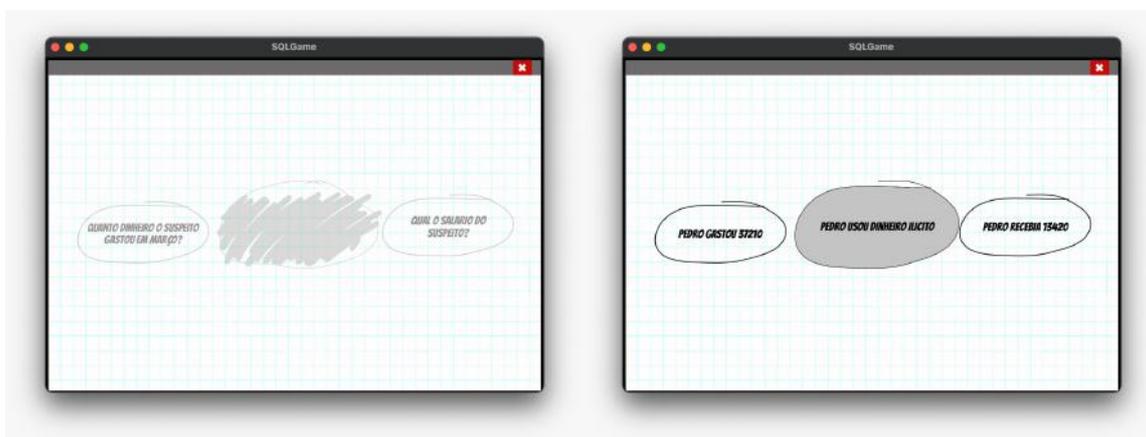
Por fim, na área inferior é onde aparecem as opções de resposta do jogador.

**Pistas** A tela de pistas, como planejado, simula um bloco de notas onde cada pista possui uma dica que é substituída pela informação após ser desvendada. As soluções/con-

Figura 20 – Tela Chat



Figura 21 – Tela Pistas



(a) Pistas

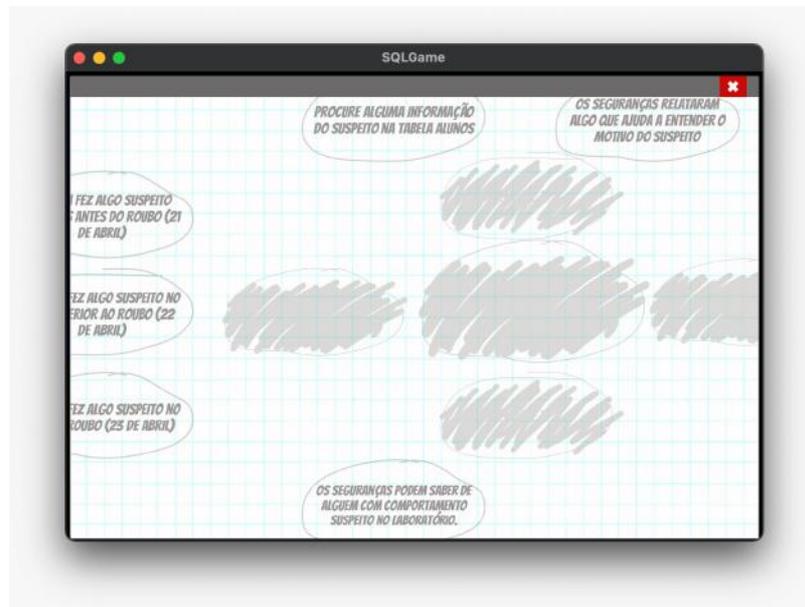
(b) Pistas encontradas

clusões são *hachuradas*, como se estivessem riscadas e só são reveladas quando as pistas às quais ela se refere são desvendadas.

A figura [21a](#) mostra como fica a tela com pistas ainda não descobertas, especificamente as pistas da fase de tutorial. As duas áreas circulares dos cantos esquerdo e direito são pistas pendentes de serem descobertas, dependendo de alguma consulta SQL. Já a área circular do meio é uma “conclusão” das duas ao seu redor, e que só será revelada quando ambas forem encontradas.

A figura [21b](#) mostra como a tela fica quando todas as pistas já foram encontradas e, analogamente, todas as conclusões liberadas.

Figura 22 – Tela Pistas mais Complexa



A área onde as pistas ficam organizadas é maior do que a tela em si, e o jogador pode arrastar a tela tanto verticalmente quanto horizontalmente, de forma a visualizar todas as pistas.

Dessa forma, conseguimos garantir uma organização lógica das pistas, ao mesmo tempo que utilizávamos uma estrutura escalonável que se aplicava tanto a fases pequenas como a da figura [21a](#) como a fases bem maiores.

A figura [22](#) mostra um exemplo de fase maior, especificamente a primeira fase, que envolve encontrar muitas pistas. Existe uma organização lógica das pistas e soluções de forma que uma determinada solução sempre tenha as pistas relacionadas a ela próximas de si.

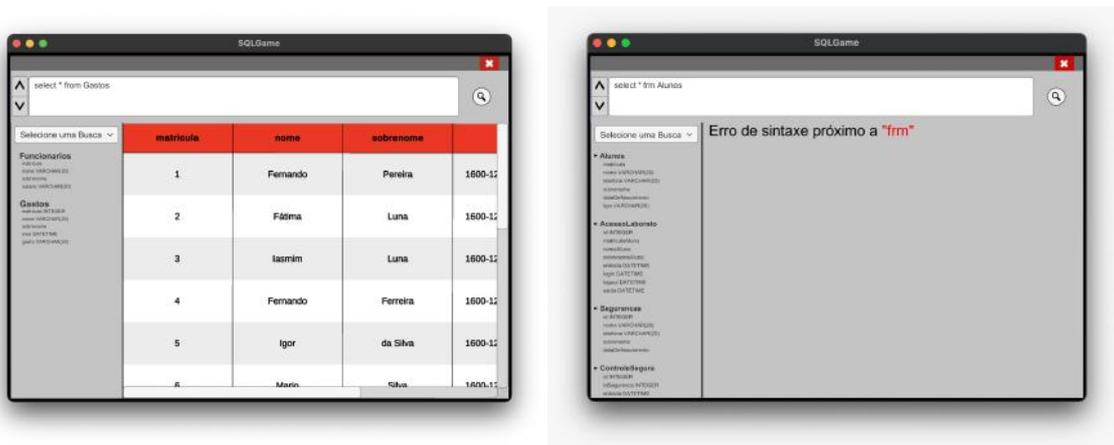
**Banco de Dados** A tela do banco de dados ficou bem semelhante à seu projeto inicial feito no Figma, porém com as correções/melhorias diagnosticadas na POC, sendo constituído basicamente de três áreas:

**Cabeçalho** que contém os botões de navegação no histórico de consultas, o campo para a escrita do comando SQL e o botão de busca que roda a consulta no banco e exibe seu resultado.

**Menu Lateral** que exibe a estrutura geral do banco de dados, com suas tabelas e colunas descritas, tal como o tipo de cada coluna, permitindo ao jogador ter uma visão ampla do banco e planejar adequadamente suas consultas.

Essa estrutura foi inicialmente planejada para ser interativa, permitindo ao jogador colapsar ou expandir as informações de uma tabela. Entretanto, devido

Figura 23 – Tela de Banco de Dados



(a) Consulta Correta

(b) Consulta com Erro

ao curto tempo, esse comportamento acabou ficando fora do escopo.

Além disso, o menu lateral possui um botão de *dropdown* com opções de *template* de consulta que o jogador pode selecionar para auxiliar na montagem das consultas que ele deseja fazer, sendo uma maneira de auxiliar em seu aprendizado da sintaxe.

**Área da Tabela** é, tal como o nome indica, a área da janela onde a tabela com os dados de retorno da consulta realizada são exibidos, permitindo um *scroll* tanto vertical quanto horizontal.

Em caso de consulta inválida, em vez de ser exibida a tabela, é mostrada uma mensagem de erro nessa mesma área, como pode ser vista na figura [23b](#).

## 5.6 CONCLUSÃO

Dessa forma, já é possível disponibilizar o jogo completo para teste, com sua história e objetivos de aprendizados inteiramente cobertos pelo MVP, avaliando, assim, se o jogo de fato pôde contribuir para o aprendizado de SQL tal qual planejado.

O código usado para preenchimento do banco de dados está disponível no apêndice D.

## 6 AVALIAÇÃO DO SQL-GAME

O jogo foi publicado com os builds (Windows, Linux e Mac) em <https://joycebrum.itch.io/sqlgame> (BRUM; OUTEIRO; XEXEO, 2022), que é uma plataforma comercial aberta para criadores de jogos independentes, com as informações necessárias para execução e avaliação do mesmo.

Figura 24 – SQL Game no itch.io



O site com os dados do jogo publicado no itch.io pode ser visto na figura 24, com alguns prints das telas do jogo também. Com isso, foi iniciado o processo de divulgação para jogadores/alunos interessados no aprendizado de SQL.

### 6.1 AVALIAÇÃO

A avaliação foi realizada com 14 pessoas e o formulário de utilizado pode ser visto no apêndice A, construído seguindo o modelo MEEGA+ previamente descrito neste documento, e os gráficos de suas respostas no apêndice B.

Avaliando primeiramente os dados demográficos coletados, podemos observar que os envolvidos na pesquisa estavam majoritariamente (75%) na faixa de 18 a 28 anos, os demais estando entre 29 e 39 anos. Além disso, a maioria dos entrevistados é do sexo masculino, e jogam jogos digitais pelo menos uma vez na semana, ou seja, estão acostumados com jogos digitais.

Metade dos entrevistados é especializado ou está se especializando em alguma área de tecnologia, os demais não possuíam, a princípio, nenhum conhecimento específico de tecnologia.

Analisando as respostas coletadas (apêndice B), calculamos a média, mediana e moda de cada dimensão e para cada pergunta. A tendência central das dimensões, a qual iremos usar para analisar o desempenho do jogo para o que havia sido proposto, pode ser visto no quadro 5

Para ver a tendência central explicitada por pergunta, vide apêndice 3.

Quadro 5 – Tendência Central das Dimensões Avaliadas

Dimensão	Média	Mediana	Moda
Estética	0,96	1	1
Aprendizibilidade	0,64	1	2
Operabilidade	0,96	1	2
Acessibilidade	1,61	1	2
Confiança	1,21	1	1
Desafio	0,52	1	1
Satisfação	1,18	1	2
Interação Social	-0,24	0	-2
Diversão	1,39	2	2
Atenção Focada	0,48	0	1
Relevância	1,38	2	2
Aprendizagem Percebida	1,07	1	2
Objetivos de Aprendizagem	1,40	2	2

Fonte: Petri, Christiane e Borgatto (2019)

As médias, medianas e modas foram coletadas após a conversão dos valores de resposta para, em vez de ir de 1 a 5, ir de -2 a 2, sendo:

**Discordo Totalmente** 1 convertido em -2

**Discordo** 2 convertido em -1

**Nem Discordo, Nem Concordo** 3 convertido em 0

**Concordo** 4 convertido em 1

**Concordo Totalmente** 5 convertido em 2

### 6.1.1 Estética

A estética é a dimensão que avalia se a interface do jogo permite uma interação agradável e satisfatória com o usuário, sendo um dos fatores importantes para medir a usabilidade do jogo (PETRI; CHRISTIANE; BORGATTO, 2019).

A média, mediana e moda do quesito estética ficaram bem próximos de 1, que seria o equivalente ao "Concordo", sendo então um critério bem avaliado.

Dessa forma, podemos entender que os jogadores acharam o design atraente (pergunta de número 9) e que os textos, cores e fontes combinavam e eram consistentes (pergunta de número 10).

Este resultado superou um pouco a expectativa que tínhamos quanto a estética do jogo levando em consideração que não tínhamos um designer no time de planejamento do mesmo.

### 6.1.2 Aprendizibilidade

Esta dimensão avalia se o jogo permite que os usuários aprendam a jogá-lo de forma fácil e rápida (PETRI; CHRISTIANE; BORGATTO, 2019).

Apesar de ter tipo uma média próxima do neutro, ao analisarmos a moda dessa dimensão vemos que a avaliação que mais se repetiu foi a de "Concordo Totalmente", mas a média indica que apesar disso houve uma quantidade razoável de avaliações entre o neutro (0) e a discordância total (-2), que acabaram diminuindo a média da amostra.

Dessa forma, percebemos que, apesar de ter sido bem avaliado houve uma parcela dos jogadores que sentiu dificuldade em aprender a jogar o jogo, o que leva a pensar em possíveis melhorias para a fase tutorial e algumas formas de explicitar melhor as mecânicas.

### 6.1.3 Operabilidade

A operabilidade é a dimensão que avalia o grau em que um jogo possui atributos que facilitam a operação e o controle (PETRI; CHRISTIANE; BORGATTO, 2019). Em outras palavras, considerando as perguntas, ele avalia se o jogo em si possui uma jogabilidade aceitável e se as regras do mesmo são claras e compreensíveis.

Os valores centrais dessa dimensão foram também muito próximos de 1, tendo sido até mais bem avaliado do que o quesito de estética considerando que, no caso da operabilidade, a moda foi 2 o que indica que a avaliação com maior ocorrência foi a de "Concordo Totalmente" nas perguntas de operabilidade.

Dessa forma, temos que a jogabilidade e as regras do jogo foram bem definidas e implementadas.

### 6.1.4 Acessibilidade

A acessibilidade avalia se o jogo pode ser jogado por pessoa com deficiência visual baixa/moderada e/ou com daltonismo (PETRI; CHRISTIANE; BORGATTO, 2019), sendo extremamente importante tendo em vista que o jogo é educativo e deveria abranger este público também.

Essa dimensão é avaliada através das seguintes critérios: fontes legíveis e cores compreensíveis (PETRI; CHRISTIANE; BORGATTO, 2019).

A avaliação dessa dimensão ficou extremamente bem colocada, com uma média de 1,61, a mais próxima de 2 dentre todas as dimensões avaliadas, e tendo como a avaliação mais frequente a de número 2.

Apesar da avaliação positiva, é preciso salientar que não há um embasamento forte na afirmação de que o jogo é acessível, pois não foi feito um teste de usabilidade com nenhum usuário que se enquadrasse em qualquer uma das deficiências supracitadas.

#### 6.1.5 Confiança

A confiança é a dimensão que avalia se os alunos são capazes de progredir no estudo do conteúdo educacional através de seu próprio esforço e habilidade (PETRI; CHRISTIANE; BORGATTO, 2019), indicando até que ponto o jogo é confiável como método de ensino.

Essa dimensão foi bem avaliada, tendo uma média de 1,21, mas tendo como mediana e moda os valores 1, indicando que apesar de se sentirem confiantes, os jogadores não afirmavam isso com total convicção.

Poderiam ser feitas melhorias no jogo para permitir que o jogador se sentisse mais confiante em sua capacidade de aprender através do jogo, como por exemplo, incluindo o conteúdo didático de alguma forma dentro do universo do mesmo.

#### 6.1.6 Desafio

O desafio é a dimensão que tem como objetivo avaliar o quanto o jogo se mostra suficientemente desafiador em relação ao nível de competência do aluno. Dessa forma, novos obstáculos e situações devem ser apresentados ao longo do jogo para minimizar a fadiga e manter os alunos interessados (PETRI; CHRISTIANE; BORGATTO, 2019).

Para medição dessa dimensão são utilizados os seguintes critérios: adequadamente desafiador, oferece novos desafios em um ritmo adequado e não se torna monótono (PETRI; CHRISTIANE; BORGATTO, 2019).

Esta dimensão foi uma das que tiveram as menores médias, tendo tido uma média de 0.52 indicando ligeiramente mais uma conclusão de concordância do que de neutralidade (nem concordando nem discordando), mas por muito pouco não obtendo o resultado oposto.

Avaliando os critérios individualmente, vimos que foram o primeiro, e o terceiro que mais impactaram negativamente a dimensão, com muitas avaliações de discordância, os quais questionavam se o jogo era adequadamente desafiador e se não se tornava monótono, respectivamente.

Quanto ao jogo ser adequadamente desafiador, o fato de haver uma disparidade de conhecimento em SQL nos envolvidos na avaliação pode ter feito com que alguns jogadores

sentissem o jogo como muito fácil e outros como adequado, levando em consideração que ele fora planejado para um jogador com conhecimentos apenas sobre fundamentos de banco de dados.

Quanto a se tornar monótono, de fato não houveram mudanças de mecânica no jogo no decorrer das fases, fazendo com que, apesar da história e mistério de cada fase serem diferentes, as atividades realizadas em cada um eram basicamente as mesmas. Para melhorar esse critério seria necessário investir em inserir novas mecânicas para quebrar a monotonia.

#### 6.1.7 Satisfação

Esta dimensão é extremamente importante, pois avalia se os jogadores sentem que seu esforço dedicado ao jogo resulta em aprendizado (PETRI; CHRISTIANE; BORGATTO, 2019).

Esta dimensão avalia se o jogador se sente realizado, se consegue avançar no jogo por seu próprio esforço, se se sente satisfeito pelo que aprendeu e se recomendaria o jogo para os seus colegas (PETRI; CHRISTIANE; BORGATTO, 2019).

Esta dimensão foi bem avaliada, tendo uma média que indica concordância com os critérios de satisfação e tendo como avaliação mais frequente a de concordância total e com a mediana 1 temos a visão de que pelo menos metade dos avaliadores concordam com os critérios de satisfação.

#### 6.1.8 Interação Social

A interação social é a dimensão que avalia se o jogo promove a sensação de um ambiente compartilhado e conexão com outras pessoas em atividades de cooperação ou competição (PETRI; CHRISTIANE; BORGATTO, 2019).

Tendo em vista que este não foi um foco do desenvolvimento do jogo, conforme esperado, esta dimensão teve uma avaliação extremamente baixa, estando com sua média entre “discordo” e “nem concordo, nem discordo”.

Sua mediana 0 indica que pelo menos metade dos avaliadores não concorda ou discorda das afirmações sobre interação social no jogo e a moda sendo -2 indica que a avaliação mais frequente foi a de “discordo totalmente”.

#### 6.1.9 Diversão

A diversão é um das dimensões principais levando em conta que o produto avaliado se trata de um jogo. Ela avalia a sensação de prazer, felicidade, relaxamento e distração dos alunos (PETRI; CHRISTIANE; BORGATTO, 2019).

Foi bem avaliada tendo tido a terceira maior média dentre os critérios avaliados, com uma média de 1,39, concluimos que as avaliações se agruparam entre “concordo” e “concordo totalmente”.

A mediana e a moda tendo o valor 2 indicam que a avaliação de “concordo totalmente” tanto foi a mais frequente quanto corresponde a pelo menos metade das respostas.

#### 6.1.10 Atenção Focada

A dimensão de atenção focada avalia a atenção, concentração focada, absorção e dissociação temporal dos jogadores enquanto jogam (PETRI; CHRISTIANE; BORGATTO, 2019), importante para auxiliar no processo de aprendizado.

Essa dimensão não foi bem avaliada, tendo suas avaliações concentradas mais entre “nem concordo, nem discordo” e “concordo”, mas a mediana 0 indicando que pelo menos metade se manteve neutro.

Essa dimensão pode ter sido afetada pelos critérios de monotonia e desafio adequado, da dimensão de desafio, que foram mal avaliados. Dessa forma, não sendo devidamente desafiado e se deparando com tarefas monótonas, o jogo não conseguiu prender tanto a atenção do jogador quanto deveria.

#### 6.1.11 Relevância

A relevância é a dimensão que avalia se os jogadores percebem que a proposta educacional do jogo é consistente com seus objetivos e que podem vincular o conteúdo ao futuro profissional ou acadêmico (PETRI; CHRISTIANE; BORGATTO, 2019).

Essa dimensão foi bem avaliada, tendo uma média alta indicando maior concentração nas avaliações de concordância, e também tendo a mediana e a moda com o valor 2, indicando que a avaliação de “concordo totalmente” tanto foi a mais frequente, quanto corresponde a pelo menos metade das avaliações feitas.

#### 6.1.12 Aprendizagem Percebida

A aprendizagem percebida avalia a percepção do efeito geral do jogo na aprendizagem dos alunos na disciplina (PETRI; CHRISTIANE; BORGATTO, 2019), usando dos critérios se o jogo contribuiu para a aprendizagem e se ele foi mais eficiente do que as outras formas de aprendizado mais tradicionais (PETRI; CHRISTIANE; BORGATTO, 2019).

Esse critério também foi bem avaliado (média, mediana e moda indicando ocorrências maiores concordância e concordâncias totais), mostrando que o jogador pôde perceber através do jogo que ele de fato estava aprendendo SQL.

### 6.1.13 Objetivos de Aprendizagem

Por fim, temos os objetivos de aprendizagem, que formam a dimensão que mede se o jogo conseguiu atingir os objetivos de aprendizado previamente definidos, ou seja, se após jogar o jogo o jogador sente que dominou ou aprendeu cada um desses objetivos (PETRI; CHRISTIANE; BORGATTO 2019).

Tais objetivos foram:

- a) Conseguir aplicar cláusulas de busca usando diferentes tabelas numa mesma consulta
- b) Lembrar as regras de sintaxe identificando os erros na escrita
- c) Analisar a estrutura do banco de dados selecionando quais informações são relevantes
- d) Criar consultas SQL usando as cláusulas necessárias
- e) Lembrar os tipos de dados identificando a forma correta de construir a query

Essa dimensão foi extremamente bem avaliada, possuindo a segunda maior média dentre as demais dimensões, com o valor 1,4 indicando concentração das avaliações entre concordância e concordância total, e tendo tanto mediana como moda em concordância total.

Dessa forma, entende-se que os objetivos de aprendizado foram adequadamente atingidos com o jogo em questão.

## 6.2 CONCLUSÃO

Tendo em vista a avaliação de cada dimensão, concluímos que o jogo conseguiu atingir os seus objetivos, sendo uma forma adequada de ensino de SQL e conseguindo entreter o jogador no processo.

Além disso, identificamos pontos de melhoria em algumas dimensões como desafio, atenção focada e interação social, sendo formas de deixar o jogo ainda mais interessante e mais atrativo.

## 7 CONCLUSÃO

Nesse ponto é importante revisarmos o que havíamos nos proposto a fazer no desenvolvimento desse jogo, primeiramente voltando ao ponto inicial do projeto: a construção do ENDO-GDC.

Vimos no tópico 2.3 como usar o ENDO-GDC para corretamente planejar um jogo educativo endógeno, cobrindo desde todas as informações relevantes para a construção do jogo em si, até todos os pontos de conteúdo didáticos que este deveria cobrir e como os mesmos seriam abordados e envolvidos no universo do jogo.

Com isso, construímos o ENDO-GDC do SQLGame no capítulo 4 definindo cada um dos objetivos de aprendizado que deveriam ser alcançados e como os mesmos seriam abordados dentro do jogo, tais foram.

- a) APLICAR as cláusulas de busca USANDO diferentes tabelas numa mesma consulta, abordado dentro do jogo a partir da fase 4 onde o jogador precisa analisar a estrutura do banco e conseguir juntar as tabelas em uma mesma consulta.
- b) LEMBRAR as regras de sintaxe IDENTIFICANDO os erros na escrita, trabalhado dentro do jogo através do botão de *dropdown* na tela de tabela, com algumas estruturas sintáticas pré-prontas e através da funcionalidade de exibição e tradução dos erros de consulta.
- c) ANALISAR a estrutura do banco de dados SELECIONANDO quais informações são relevantes para a fase, explorado no jogo através do menu lateral da tela de tabela, onde é possível para o jogador ver as tabelas da fase e suas colunas, analisando quais informações poderiam ser relevantes para obter as pistas remanescentes.
- d) CRIAR consultas em SQL CONSTRUINDO a query usando as cláusulas necessárias é um objetivo que é explorando o jogo inteiro, sendo a mecânica de “encontrar uma pista”.
- e) LEMBRAR os tipos de dados IDENTIFICANDO as formas corretas de construir a *query* é viabilizado no jogo através do menu lateral da tela de tabela, que mostra os tipos de cada coluna das tabelas do banco, além de algumas fases exigirem filtros por tipos como data e trecho de string.

Com a avaliação final do jogo, foi possível perceber que os jogadores sentiram que de fato aprenderam tais objetivos de aprendizado através do jogo, além de terem conseguido se divertir e se envolver com o universo do mesmo.

Mas, apesar do resultado positivo, tiveram alguns pontos que poderiam ser trabalhados para aumentar a qualidade do jogo final, cobrindo dimensões que não foram bem avaliadas pelos jogadores: atenção focada, desafio, interação social e aprendizibilidade.

Com isso, um dos possíveis trabalhos futuros para este projeto seria explorar algumas melhorias nessas 4 dimensões, sendo algumas ideias listadas abaixo.

- Permitir acesso ao conteúdo didático dentro do jogo, para aumentar a imersão evitando que o jogador precise sair do mesmo para obter esse conteúdo;
- Melhorar a fase de tutorial para ser mais interativa, com suas instruções cobrindo a resolução completa da fase;
- Adicionar novas mecânicas no decorrer das fases, diminuindo a monotonia, tais quais: mini jogos para “descriptografar” tabelas do banco, conquistas secretas que o jogador pode obter ao explorar o jogo, etc;
- Cobrir com o jogo mais conteúdos didáticos de SQL, tais como: DDL, DML, DCL e DTL.

## REFERÊNCIAS

- ABT, C. C. **Serious Games**. New York: Lanham, 2002. ISBN 978-0-8191-6148-2.
- ALVES, E. Jogos sérios para ensino de engenharia de software. 2013.
- ATLASSIAN. **O Trello facilita a gestão de projetos e tarefas para os times**. 2022. Disponível em: <https://trello.com/pt-BR/tour>.
- BATISTA, A. L. F. et al. Sql planet: Jogo online para ensino de linguagem sql. **SBC**, 2019. Disponível em: <https://www.sbgames.org/sbgames2019/files/papers/EducacaoShort/197137.pdf>. Acesso em: 26 jul.2020.
- BEAULIEU, A. **Learning SQL, second edition**. [S.l.]: O'REILLY, 2009. 9 p.
- BEIGHLEY, L. **Use a Cabeça! SQL**. [S.l.]: O'Reilly, 2010. ISBN 978-85-7608-210-1.
- BRUM, J.; OUTEIRO, T.; XEXEO, G. **SQLGame**. 2022. Disponível em: <https://joycebrum.itch.io/sqlgame>
- CAMPBELL, J. **The Hero with a Thousand Faces**. [S.l.]: Pantheon Books, 1949.
- CHAND, M. **Most Popular Databases In The World**. 2022. Disponível em: <https://www.c-sharpcorner.com/article/what-is-the-most-popular-database-in-the-world/>.
- DILLON, R. The 6-11 framework: A new methodology for game analysis and design. **DigiPen Institute of Technology, Singapore**, 2011.
- ECK, R. V. **Digital Game-Based Learning: It's Not Just the Digital Natives Who Are Restless**. EDUCASE, 2006. Disponível em: <https://er.educause.edu/articles/2006/1/digital-gamebased-learning-its-not-just-the-digital-natives-who-are-restless>. Acesso em: 17 mar.2022.
- EGENFELDT-NIELSEN, S. **Beyond edutainment: Exploring the educational potential of computer games**. [S.l.]: Lulu. com, 2011.
- GAGNÉ, R. M.; WAGNER, W. W.; GOLAS, K. e. a. **Principles of Instructional Design 5th edition**. [S.l.]: Cengage Learning, 2004. ISBN 978-0-534-58284-5.
- GARNER, P.; MARIANI, J. Learning sql in steps. **SYSTEMICS, CYBERNETICS AND INFORMATICS VOLUME 13 - NUMBER 4 - YEAR 2015**, 2015. Disponível em: <http://www.iiisci.org/journal/pdv/sci/pdfs/EB123GK15.pdf>.
- GERMUSKA, J. **SQL Murder Mystery**. 2019. Disponível em: <https://github.com/NUKnightLab/sql-mysteries>.
- GRUPMAN, C. **Why Learn SQL in 2022 (Analyzing Real Job Data)**. 2021. Disponível em: <https://www.dataquest.io/blog/why-sql-is-the-most-important-language-to-learn/>
- IMDB. **Her Story**. 2021. Disponível em: <https://www.imdb.com/title/tt5438010/awards>.

LEWIS, M.; JACOBSON, J. Game engine in scientific research. **Communication of the ACM**, v. 45, Janeiro 2002.

LIEBERMAN, D. A. What can we learn from playing interactive games. **Playing Video Games: Motives, Responses, and Consequences (pp.379-397)**, 2006.

LIMA, V. **As 5 melhores engines para criação de jogos**. 2018. Disponível em: <https://blog.schoolofnet.com/as-5-melhores-engines-para-criacao-de-jogos/>.

MCGONIGAL, J.; RIECHE, E. **A realidade em jogo: Por que os games nos tornam melhor e como eles podem mudar o mundo**. [S.l.: s.n.], 2012. ISBN 978-85-7684-522-5.

MICROSOFT. **Documentação Técnica do SQL Server**. 2022. Disponível em: <https://docs.microsoft.com/pt-br/sql/sql-server/?view=sql-server-ver15>.

MYSQL. **MySQL Documentation**. 2022. Disponível em: <https://dev.mysql.com/doc/>.

ORACLE. **Oracle Database Online Documentation 10g Release 2 (10.2)**. 2015. Disponível em: [https://docs.oracle.com/cd/B19306\\_01/index.htm](https://docs.oracle.com/cd/B19306_01/index.htm).

PETRI, G.; CHRISTIANE, W.; BORGATTO, A. Meega+: Um modelo para a avaliação de jogos educacionais para o ensino de computação. **Revista Brasileira de Informática na Educação**, Setembro 2019.

SANTOS, E.; RIBEIRO, P. Ludosql: Jogo de apoio ao aprendizado de comandos básicos do sql. **Faculdade de Tecnologia de Bauru**, 2016.

STEAM. **Orwell: Keeping an Eye On You**. 2016. Disponível em: [https://store.steampowered.com/app/491950/Orwell\\_Keeping\\_an\\_Eye\\_On\\_You/](https://store.steampowered.com/app/491950/Orwell_Keeping_an_Eye_On_You/). Acesso em: 27 out.2016.

STEAMSPY. **Orwell: Keeping an Eye On You**. 2016. Disponível em: <https://steamspy.com/app/491950>. Acesso em: 17 abr.2022.

STEAMSPY. **App - Her Story**. 2021. Disponível em: <https://steamspy.com/app/368370>. Acesso em: 26 jul.2021.

STUDIOS, O. **Orwell: Keeping an Eye on You**. 2018. Disponível em: <https://www.osmoticstudios.com/orwell-keeping-an-eye-on-you/>. Acesso em: 17 abr.2022.

SUSI, T.; JOHANNESSON, M.; BACKLUND, P. Serious games: An overview. 2007.

TAUCEI, B. B. Endo-gdc: Desenvolvimento de um game design canvas para concepção de jogos educativos endógenos. **Universidade Federal do Rio de Janeiro**, 2019. Disponível em: <https://www.cos.ufrj.br/uploadfile/publicacao/2928.pdf>. Acesso em: 26 jul.2020.

TAYLOR, A. G. **SQL Para Leigos**. [S.l.]: Alta Books, 2016. ISBN 978-85-7608-967-4.

TYLER, D. **How to Choose the Best Video Game Engine**. 2022. Disponível em: <https://www.gamedesigning.org/career/video-game-engines/>.

VOGLER, C. **The Writer's Journey: Mythic Structures for Storytellers & Screenwriters**. [S.l.]: Michael Wiese Productions, 2007.

WINN, B. The design, play, and experience framework. **Handbook of research on effective electronic gaming in education**, p. 1010–1024, 2009.

XEXEO, G. et al. O que são jogos, uma introdução ao objeto de estudo do ludes. 2017. Disponível em: <<https://www.cos.ufrj.br/uploadfile/publicacao/2766.pdf>>.

ZYDA, M. From visual simulation to virtual reality to games. **Computer**, 2005. Disponível em: <<http://dx.doi.org/10.1109/MC.2005.297>>. Acesso em: 28 jul 2020.

## APÊNDICES

### APÊNDICE A – AVALIAÇÃO DISCENTE: SQLGAME, JOGO EDUCACIONAL PARA ENSINO DE SQL

1. Você concorda em participar da pesquisa "uso do jogo SQLGame para ensino SQL", de acordo com o o Termo de Consentimento Livre e Esclarecido? (sim ou não)
2. Nome (Texto simples)
3. Email (Texto simples)
4. Faixa Etária
  - a) Menos de 18 anos
  - b) 18 a 28 anos
  - c) 29 a 39 anos
  - d) 40 a 50 anos
  - e) Mais de 50 anos
5. Sexo
  - a) Feminino
  - b) Masculino
  - c) Não Declarar
6. Que curso você faz/se graduou? (Texto Simples)
7. Com que frequência você costuma jogar jogos digitais
  - a) Nunca: nunca jogo
  - b) Raramente: Jogo de tempos em tempos
  - c) Mensalmente: jogo pelo menos uma vez por mês
  - d) Semanalmente: jogo pelo menos uma vez por semana
  - e) Diariamente: jogo todos os dias
8. Com que frequência você costuma jogar jogos não-digitais (de cartas, tabuleiro, etc)
  - a) Nunca: nunca jogo
  - b) Raramente: Jogo de tempos em tempos
  - c) Mensalmente: jogo pelo menos uma vez por mês
  - d) Semanalmente: jogo pelo menos uma vez por semana
  - e) Diariamente: jogo todos os dias







36. O jogo é um método de ensino adequado para esta disciplina.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

37. Eu prefiro aprender com este jogo do que de outra forma (outro método de ensino).

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

38. O jogo contribuiu para a minha aprendizagem na disciplina.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

39. O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

40. O jogo contribuiu para aplicar as cláusulas de busca usando diferentes tabelas numa mesma consulta.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

41. O jogo contribuiu para lembrar as regras de sintaxe identificando os erros na escrita.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

42. O jogo contribuiu para analisar a estrutura do banco de dados selecionando quais informações são relevantes para a fase.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

43. O jogo contribuiu para criar consultas em SQL construindo a query usando as cláusulas necessárias.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

44. O jogo contribuiu para lembrar os tipos de dados identificando as formas corretas de construir a query.

1      2      3      4      5  
 Discordo Totalmente    ○    ○    ○    ○    ○    Concordo Totalmente

## APÊNDICE B – RESPOSTAS DO FORMULÁRIO DE AVALIAÇÃO

As respostas para cada pergunta podem ser vistas a baixo. As figuras [25](#), [26](#), [27](#), [28](#), [29](#) e [30](#) possuem menos respostas pois foram adicionadas ao formulário após algumas avaliações já terem sido feitas.

Figura 25 – Resposta Questionário: Autorização de Participação

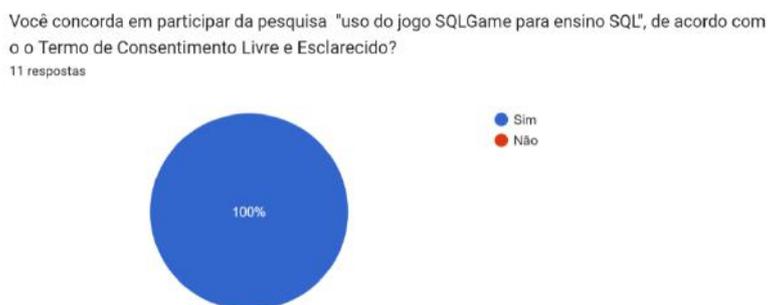


Figura 26 – Resposta do Questionário: Faixa Etária

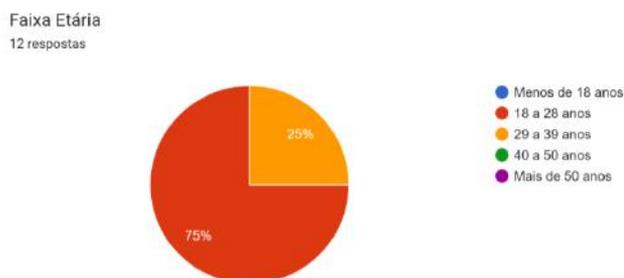


Figura 27 – Resposta do Questionário: Sexo

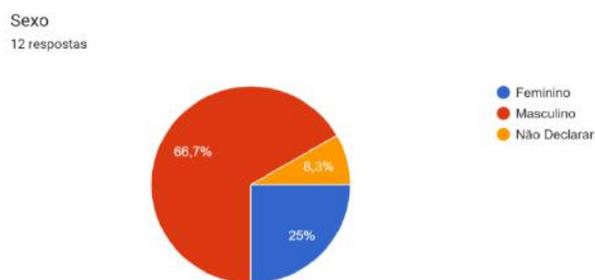


Figura 28 – Resposta do Questionário: Curso de Graduação

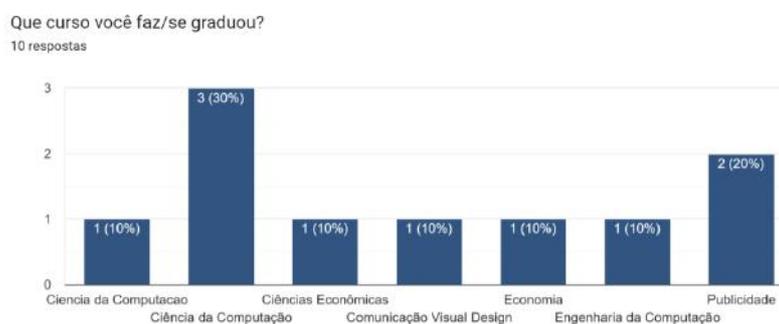


Figura 29 – Resposta do Questionário: Frequência de Jogos Digitais

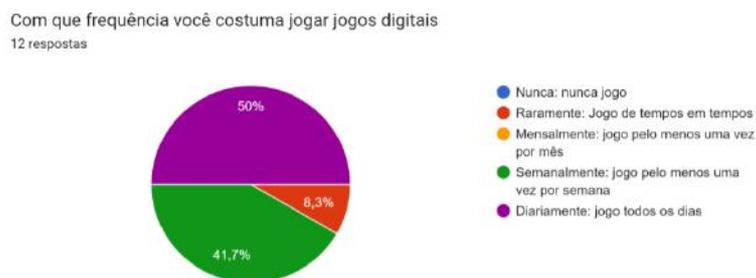


Figura 30 – Resposta do Questionário: Frequência de Jogos não Digitais

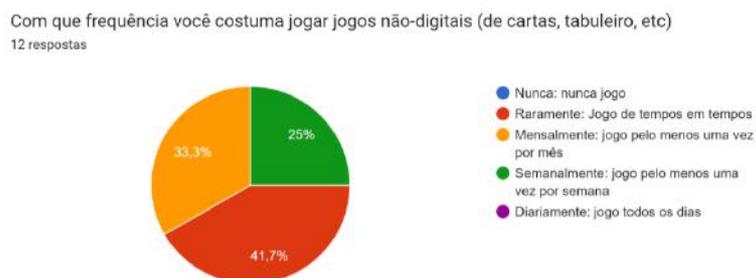


Figura 31 – Resposta do Questionário: Design do Jogo Atraente

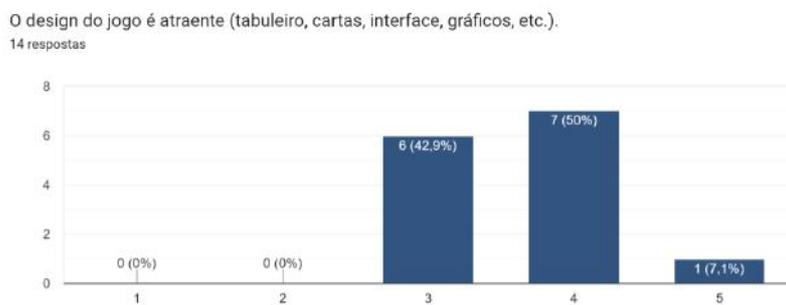


Figura 32 – Resposta do Questionário: As Cores Combinam

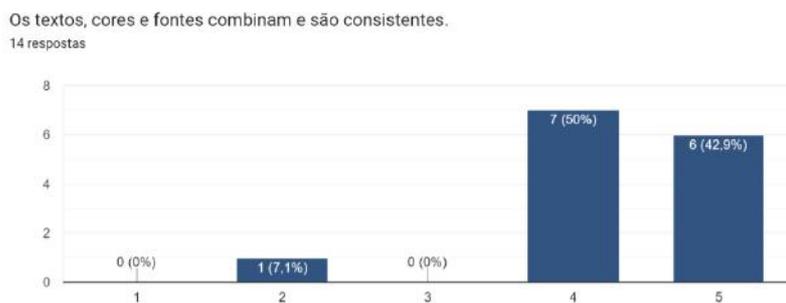


Figura 33 – Resposta do Questionário: Rápido de Aprender

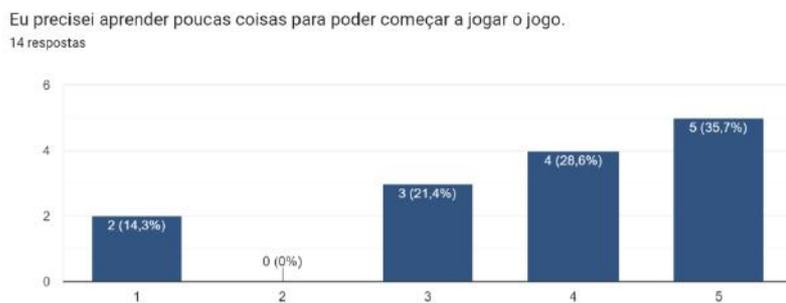


Figura 34 – Resposta do Questionário: Fácil de Aprender

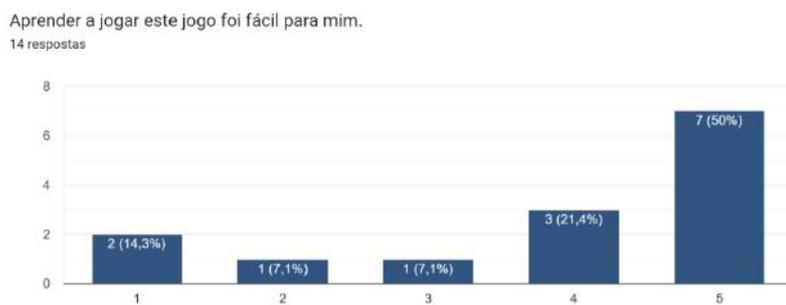


Figura 35 – Resposta do Questionário: Outros Aprenderiam Fácil



Figura 36 – Resposta do Questionário: Fácil de Jogar

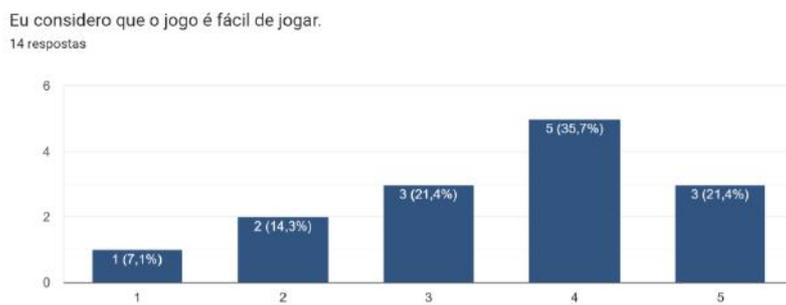


Figura 37 – Resposta do Questionário: Regras Claras

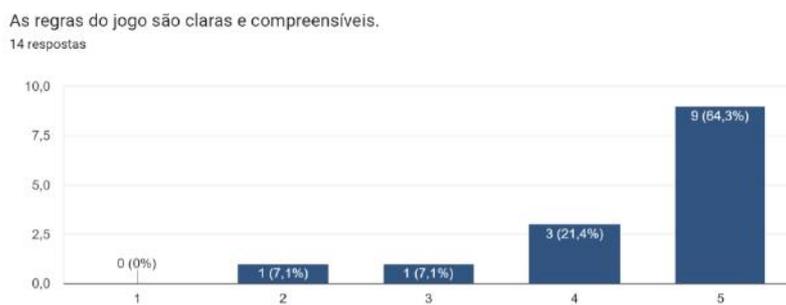


Figura 38 – Resposta do Questionário: Fontes Legíveis



Figura 39 – Resposta do Questionário: Cores Compreensíveis

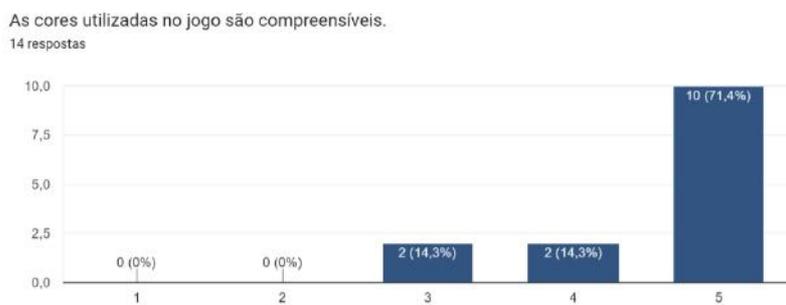


Figura 40 – Resposta do Questionário: Confiante sobre Aprender

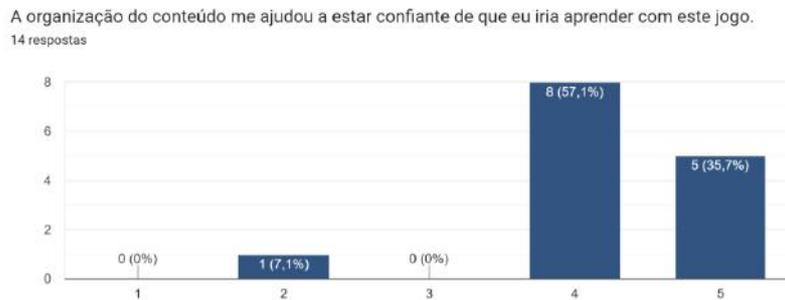


Figura 41 – Resposta do Questionário: Adequadamente Desafiador

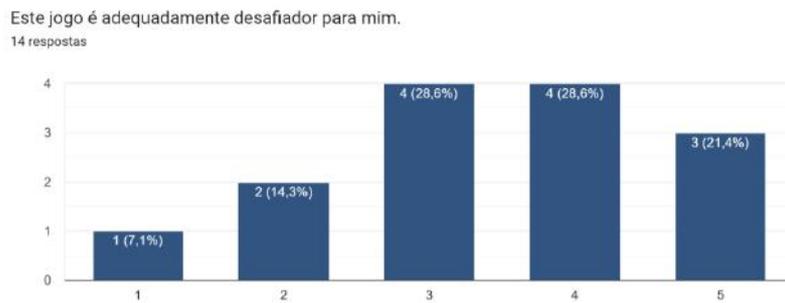


Figura 42 – Resposta do Questionário: Novos Desafios

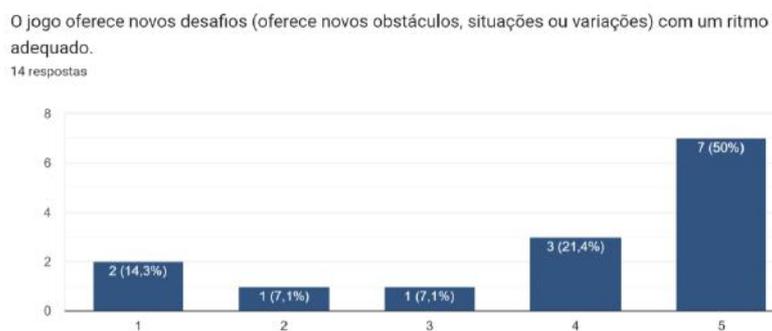


Figura 43 – Resposta do Questionário: Não Monótono

O jogo não se torna monótono nas suas tarefas (repetitivo ou com tarefas chatas).

14 respostas

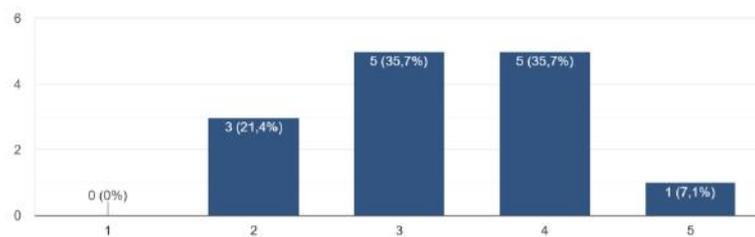


Figura 44 – Resposta do Questionário: Realização

Completar as tarefas do jogo me deu um sentimento de realização.

14 respostas

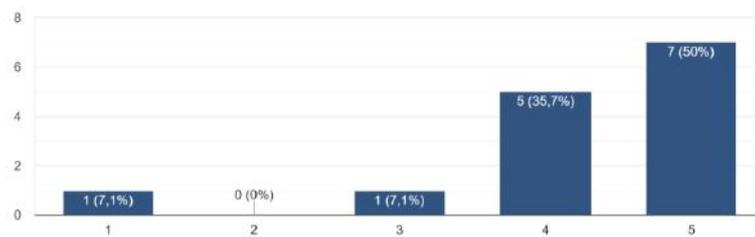


Figura 45 – Resposta do Questionário: Esforço Pessoal

É devido ao meu esforço pessoal que eu consigo avançar no jogo.

14 respostas

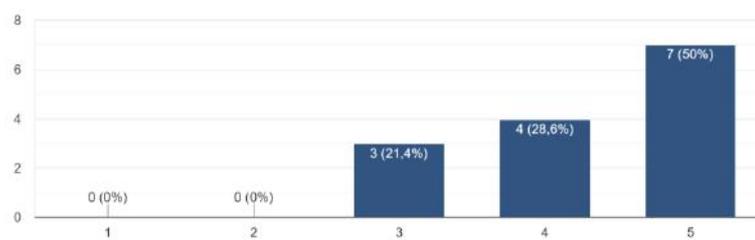


Figura 46 – Resposta do Questionário: Satisfação

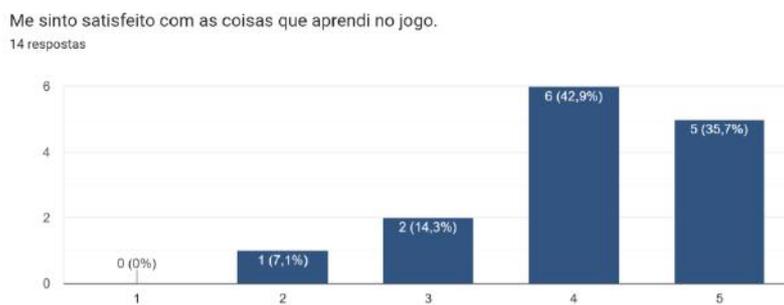


Figura 47 – Resposta do Questionário: Recomendação

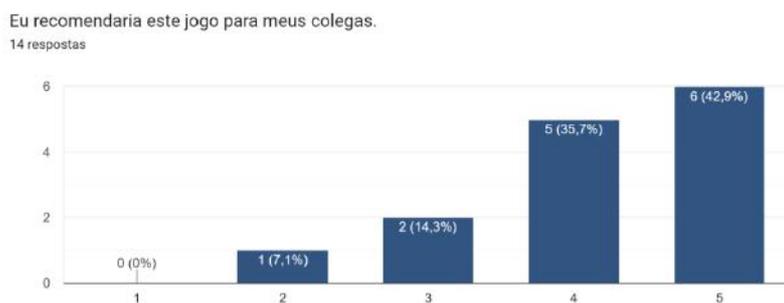


Figura 48 – Resposta do Questionário: Interação Social

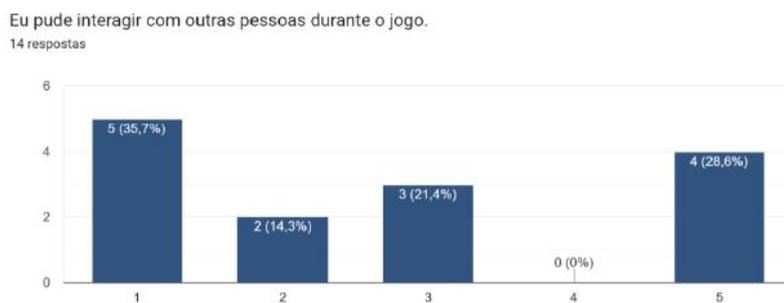


Figura 49 – Resposta do Questionário: Cooperação/Competição

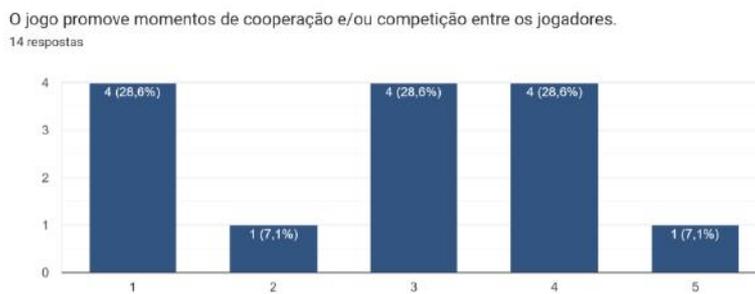


Figura 50 – Resposta do Questionário: Boa Interação

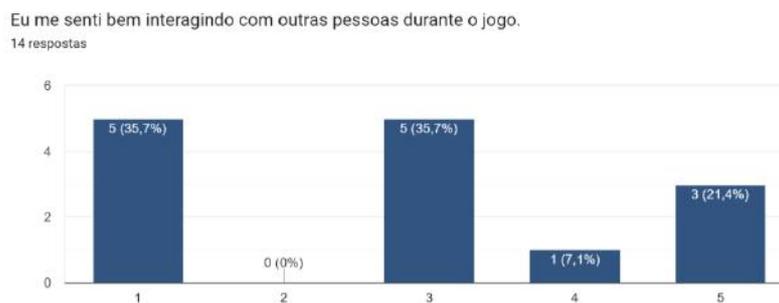


Figura 51 – Resposta do Questionário: Diversão

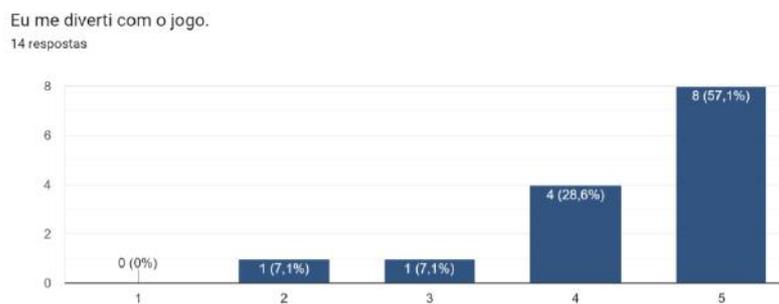


Figura 52 – Resposta do Questionário: Alegria

Aconteceu alguma situação durante o jogo (elementos do jogo, competição, etc.) que me fez sorrir.

14 respostas

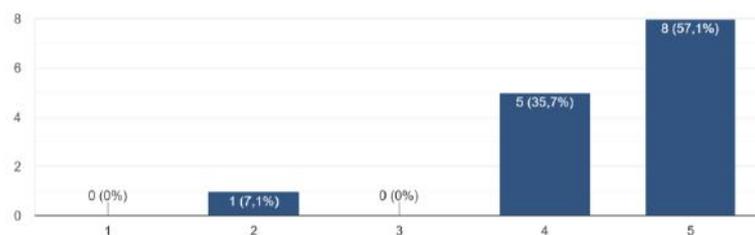


Figura 53 – Resposta do Questionário: Interesse

Houve algo interessante no início do jogo que capturou minha atenção.

14 respostas

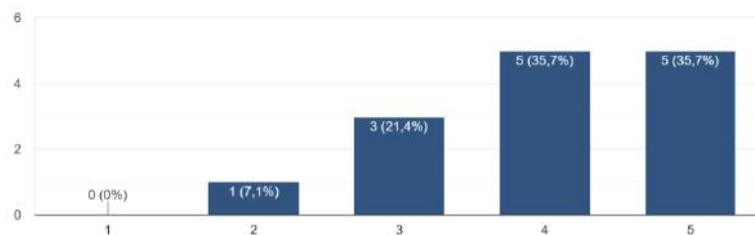


Figura 54 – Resposta do Questionário: Imersão - Noção do tempo

Eu estava tão envolvido no jogo que eu perdi a noção do tempo.

14 respostas

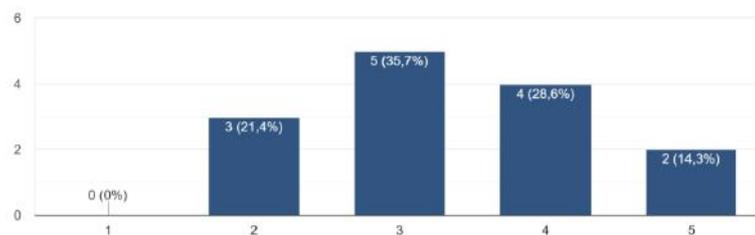


Figura 55 – Resposta do Questionário: Imersão - Esquecer do ambiente

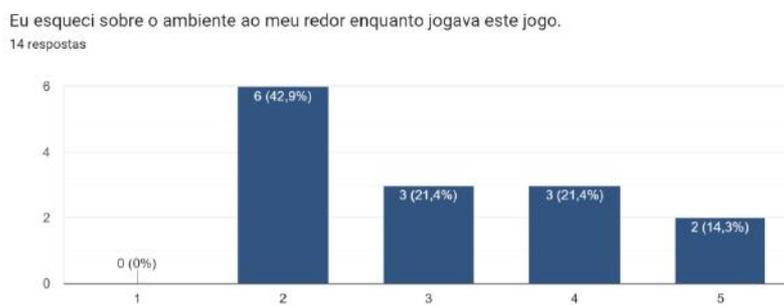


Figura 56 – Resposta do Questionário: Relevância

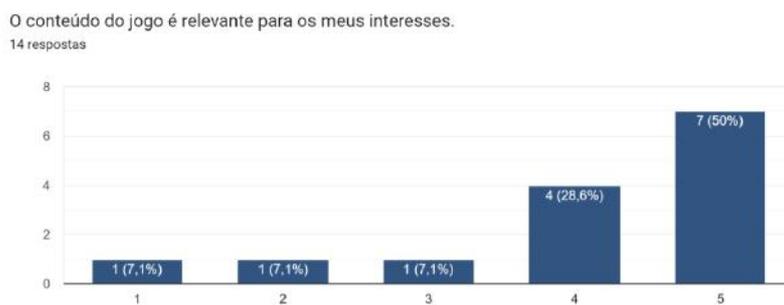


Figura 57 – Resposta do Questionário: Relacionado a SQL



Figura 58 – Resposta do Questionário: Bom Método



Figura 59 – Resposta do Questionário: Preferência

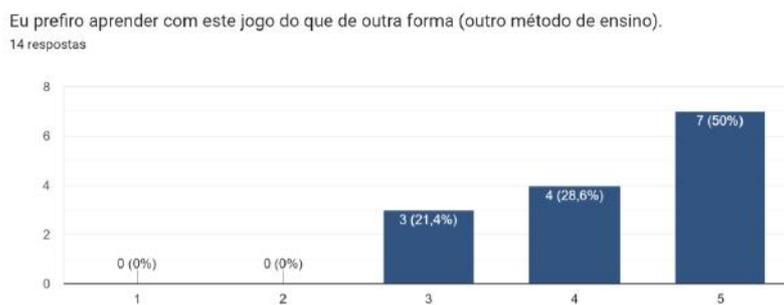


Figura 60 – Resposta do Questionário: Contribuição na Aprendizagem



Figura 61 – Resposta do Questionário: Eficiência na Aprendizagem



Figura 62 – Resposta do Questionário: Objetivo de Aprendizado 1

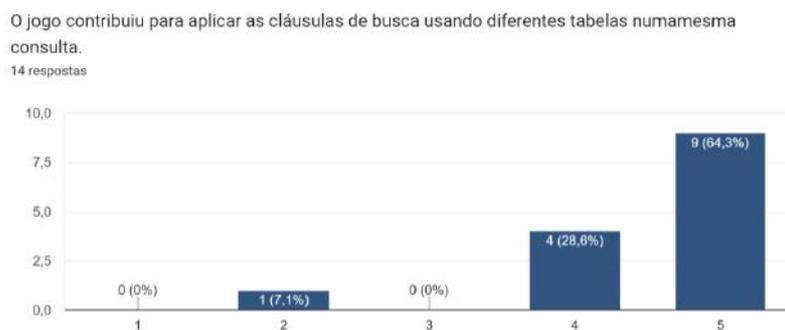


Figura 63 – Resposta do Questionário: Objetivo de Aprendizado 2



Figura 64 – Resposta do Questionário: Objetivo de Aprendizado 3

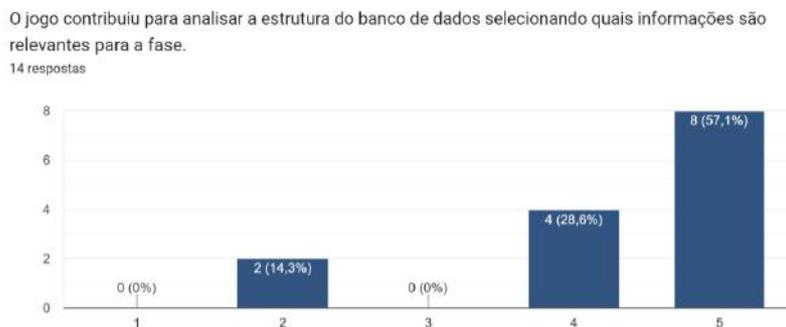
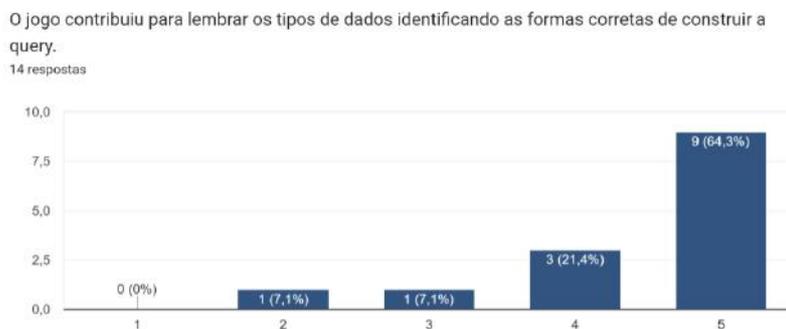


Figura 65 – Resposta do Questionário: Objetivo de Aprendizado 4



Figura 66 – Resposta do Questionário: Objetivo de Aprendizado 5



## APÊNDICE C – TENDÊNCIA CENTRAL DE CADA PERGUNTA

Quadro 6 – Tendência Central das Perguntas

Pergunta	Média	Mediana	Moda
9	0,64	1	1
10	1,29	1	1
11	0,71	1	2
12	0,86	1	2
13	0,36	0	1
14	0,5	1	1
15	1,43	2	2
16	1,64	2	2
17	1,57	2	2
18	1,21	1	1
19	0,43	0	0
20	0,86	1	2
21	0,29	0	1
22	1,21	1	2
23	1,29	1	2
24	1,07	1	1
25	1,14	1	2
26	-0,29	-1	-2
27	-0,21	0	0
28	-0,21	0	0
29	1,36	2	2
30	1,43	2	2
31	1	1	1
32	0,36	0	0
33	0,07	0	-1
34	1,07	1	2
35	1,86	2	2
36	1,29	2	2
37	1,29	1	2
38	0,86	1	2
39	1,29	1	2
40	1,5	2	2
41	1,43	2	2
42	1,29	2	2
43	1,36	2	2
44	1,43	2	2

Fonte: Petri, Christiane e Borgatto (2019)

## APÊNDICE D – SCRIPT PARA GERAÇÃO DAS FASES

### D.1 FASE 1

Listing D.1 – Geração SQL Tabelas

```

import random
import datetime

# Metodos de geracao aleatoria

nomes = [
    "Andre", "Alice", "Alfredo", "Arthur", "Armando", "Adriana", "
    Adriano", "Alberto",
    "Beatriz", "Bianca", "Breno", "Bruno", "Bruna", "Bob", "Bernardo", "
    Benjamin", "Bento", "Berenice", "Barney", "Barbara",
    "Caio", "Caroline", "Carolina", "Carlor", "Cauan", "Cleberson", "
    Cicero", "Cecilia", "Davi", "Camila",
    "Eduardo", "Eduarda", "Estevam", "Elena", "Elenilce", "Ester", "
    Everton", "Elen", "Elizandra", "Elizete",
    "Fabio", "Fabiana", "Fernando", "Fernanda", "Fatima", "Felipe", "
    Flavia",
    "Gabriel", "Gabriela", "Gustavo", "Galvao", "Giovani", "Giovana", "
    Gerson",
    "Hugo", "Helena", "Helen",
    "Igor", "Iasmim", "Ivan", "Isaac", "Ivone",
    "Joao", "Joana", "Jonas", "Julio", "Juliana", "Josue", "Jhonatan", "
    Jardel", "Janaina", "Joyce", "Joseli", "Joselaine", "Jasmine", "
    Juan", "Jade",
    "Kerolin", "Kauan",
    "Leticia", "Lucas", "Luan", "Lilian", "Laura", "Leonardo",
    "Marcos", "Maicon", "Marcelo", "Marcela", "Miriam", "Mauricio", "
    Mario", "Mariana",
    "Natan", "Natalia",
    "Orlando", "Oscar", "Osmar",
    "Pedro", "Pietro", "Paula", "Paulo", "Pamela", "Paloma", "Patricia",
    "Patricio", "Penelope", "Priscila",
    "Renato", "Renata", "Rodrigo", "Rubens", "Ricardo", "Roberta",
    "Susan", "Savio", "Suelen", "Silmar",
    "Tuane", "Thiago", "Tatiana",
    "Victor", "Viviane", "Vivian", "Velma", "Valkiria",
    "Yasmim"
]

sobrenomes = [

```

```
"Sousa", "Souza", "Sant Anna", "Brum", "Goncalves", "Ferreira", "
    Nascimento", "do Nascimento", "Santos", "dos Santos", "de Sant
    Anna", "Ribeiro", "Pereira",
"Outeiro", "Campos", "Matos", "Silva", "da Silva", "Arnaud", "Scholl
    ", "Rodrigues", "Carvalho", "Luna", "Lima", "de Lima", "de
    Carvalho", "Gomes", "Oliveira",
"Alves", "Martins", "Cerqueira", "Barros", "Barbosa", "Andrade", "
    Batista", "Cardoso", "Castro", "Costa", "Dias", "Duarte", "
    Freitas", "Garcia", "Lopes", "Felix",
"Fernandes", "Machado", "Marques", "Medeiros", "Melo", "Monteiro", "
    Moreira", "Nunes", "Oliveira", "Moura", "Reis", "Rocha", "
    Teixeira", "Vieira"]
```

```
matriculas_usadas = {'152354': True, '155623': True, '163224': True}
```

```
def nova_matricula():
```

```
    value = random.randrange(150000, 200000)
    while str(value) in matriculas_usadas:
        value = random.randrange(150000, 200000)
    matriculas_usadas[str(value)] = True
    return str(value)
```

```
def rand_tipo():
```

```
    tipos = ['ampla', 'baixa renda', 'racial']
    return tipos[random.randrange(0, len(tipos))]
```

```
def rand_telephone():
```

```
    ddds = [11, 12, 13, 14, 15, 16,      17,      18, 19, 21,      22,
            24,      27,      28,      31,      32,      33,      34,
            35, 37, 38, 41,      42,      43,      44,      45,      46,
            47,      48,      49,      51,      53,      54,      55,
            61,      62,
            63, 64,      65, 66, 67,      68,      69,      71,      73,
            74,      75,      77,      79,      81,      82,      83,
            84,      85,
            86, 87,      88,      89,      91,      92,      93,      94,
            95,      96,      97, 98, 99]
    return "(" + str(ddds[random.randrange(0, len(ddds))]) + ") " + str(
        random.randrange(1000, 99999)) + "-" + str(random.randrange(1000,
        9999))
```

```
def rand_data_nascimento():
```

```
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11",
        ", "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10",
        ", "11", "12"]
```

```

dia = random.randrange(1, 31)

if dia == 31:
    mes = meses31[random.randrange(0, len(meses31))]
elif dia > 28:
    mes = meses30[random.randrange(0, len(meses30))]
else:
    mes = meses28[random.randrange(0, len(meses28))]
ano = random.randrange(1994, 2003)
return str(ano) + "-" + mes + "-" + str(dia)

def rand_data_nascimento_seguranca():
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11",
               ", "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10",
               ", "11", "12"]
    dia = random.randrange(1, 31)

    if dia == 31:
        mes = meses31[random.randrange(0, len(meses31))]
    elif dia > 28:
        mes = meses30[random.randrange(0, len(meses30))]
    else:
        mes = meses28[random.randrange(0, len(meses28))]
    ano = random.randrange(1974, 1990)
    return str(ano) + "-" + mes + "-" + str(dia)

dataInicial = datetime.datetime(2022, 4, 1)

def rand_access(dia, mes, maxH):
    hora = random.randrange(7, 20)
    minuto = random.randrange(0, 59)
    segundo = random.randrange(0, 59)

    duracaoH, duracaoM, duracaoS = random.randrange(0, maxH), random.
        randrange(0, 59), random.randrange(0, 59)
    fsegundo = (segundo + duracaoS) % 60
    fminuto = (minuto + duracaoM + int((segundo + duracaoS) / 60)) % 60
    fhora = hora + duracaoH + int((minuto + duracaoM) / 60)
    if fhora > 20: fhora = 20
    return datetime.datetime(2022, mes, dia, hora, minuto, segundo),
        datetime.datetime(2022, mes, dia, fhora, fminuto, fsegundo)

# Dados de Materias - responsavel por preencher a tabela "Materias" e
    salvar dados de materias

```

```

titulos = [
    "Computadores e Sociedade", "Seguranca da Informacao", "Computacao I
    ", "Computacao II", "Computacao III", "Numeros Inteiros e
    Criptografia", "Calculo I",
    "Calculo II", "Calculo III", "Calculo IV", "Projeto de Jogos", "
    Engenharia de Software", "Inteligencia Artificial", "
    Desenvolvimento Web", "Banco de Dados I",
    "Banco de Dados II", "Computacao Concorrente", "Computacao Paralela"
    , "Sistemas Distribuidos", "Algoritmos e Grafos", "Logica", "
    Analise de Desempenho",
    "Computacao Grafica", "Probabilidade e Estatistica", "Engenharia de
    Sistemas", "Fisica I", "Fisica II" , "Fisica III", "Fisica IV"]

print("INSERT INTO Materias(id, titulo) VALUES ")

count = 0
length = len(titulos)
materias = []
for titulo in titulos:
    _id = count + 1
    materias.append( (_id, titulo) )
    if count == length - 1:
        print("    (" + str(_id) + ", '" + titulo + "');")
    else:
        print("    (" + str(_id) + ", '" + titulo + "'),")
    count += 1

print("\n")

# Dados dos Seguranças - gera aleatoriamente e guarda seguranca por
turno para referencia posterior

class Seguranca:
    def __init__(self, id=None):
        self.randomize(id)

    def randomize(self, id):
        self.id = id
        self.name = random.choice(nomes)
        self.last_name = random.choice(sobrenomes)
        self.tel = rand_telephone()
        self.birthdate = rand_data_nascimento_seguranca()
        self.last = False

    def is_last(self):
        self.last = True

```

```

def __str__(self):
    ending = ';' if self.last else ','
    return " (" + str(self.id) + ", '" + self.name + "', '" +
        self.last_name + "', '" + self.tel + "', '" + self.birthdate
        + "')" + ending

def generate_security_shift(total, last_id):
    security_shift = []
    for i in range(0, total):
        last_id += 1
        security_shift.append(Seguranca(last_id))
    return security_shift, last_id

security_m, last_id = generate_security_shift(20, 0)      # segurancas
                 do turno da manha de 7 as 15

security_t, last_id = generate_security_shift(20, last_id) # segurancas
                 do turno da tarde de 13 as 21

print("INSERT INTO Segurancas(id, nome, sobrenome, telefone,
      dataDeNascimento) \nVALUES")

for security in security_m:
    print(security);

for i in range(0, len(security_t)):
    security = security_t[i]
    if i == len(security_t) - 1:
        security.is_last()
    print(security)

print ("\n")

# Dados de Alunos - preenche aleatoriamente a tabela Alunos e salva para
# referencia posterior

print("INSERT INTO Alunos(matricula, nome, sobrenome, telefone,
      dataDeNascimento, tipo) \nVALUES")

count = 0
total = 200
ending = ','

alunos = [

```

```

('152354', 'Caio', 'Bezerra', rand_telephone(), rand_data_nascimento
(), 'baixa renda'),
('155623', 'Vitor', 'Damasceno', rand_telephone(),
rand_data_nascimento(), 'ampla'),
('163224', 'Thiago', 'Miloni', rand_telephone(),
rand_data_nascimento(), 'baixa renda'),
]

for i in range(0, total):
    aluno = (nova_matricula(), random.choice(nomes), random.choice(
sobrenomes), rand_telephone(), rand_data_nascimento(), rand_tipo
())
    alunos.append(aluno)

for i in range(0, len(alunos)):
    aluno = alunos[i]
    if i == len(alunos)-1:
        ending = ';'
    print("      ('+ aluno[0] + "', ' + aluno[1] + "', ' + aluno[2] + "
', ' + aluno[3] + "', ' + aluno[4] + "', ' + aluno[5] + "')" +
ending)

print("\n")

# Materias por Aluno - preenche a tabela MateriasPorAluno, relacionando
alunos e suas respectivas materias

print("INSERT INTO MateriasPorAluno(idMateria, nomeMateria,
matriculaAluno, nomeAluno) \nVALUES")

tempMaterias = materias.copy()
quantMateria = {}
for aluno in alunos:
    materia = tempMaterias[random.randrange(0, len(tempMaterias))]
    if materia[1] in quantMateria:
        quantMateria[materia[1]] += 1
    else:
        quantMateria[materia[1]] = 0
    if quantMateria[materia[1]] > 40:
        tempMaterias.remove(materia[1])

    print("      (" + str(materia[0]) + ", ' + materia[1] + "', " + str(
aluno[0]) + ", ' + aluno[1] + "')")

print("\n")

```

```

# Acesso aos Laboratorios - gera acessos randomicos para os alunos ao
laboratorio ate a data do roubo

def generateCaio(dia):
    if dia == 21:
        print("      ('152354', 'Caio', 'Bezerra', '2022-04-21 13:20:07',
                '2022-04-21 15:00:15', '2022-04-21 17:06:23', '2022-04-21
                21:48:37'),")
    if dia == 22:
        print("      ('152354', 'Caio', 'Bezerra', '2022-04-22 13:50:44',
                '2022-04-22 15:10:41', '2022-04-22 19:49:27', '2022-04-23
                21:15:20'),")
    if dia == 23:
        print("      ('152354', 'Caio', 'Bezerra', '2022-04-23 12:59:01',
                '2022-04-23 14:03:16', '2022-04-23 18:19:59', '2022-04-23
                21:37:48'),")

print("INSERT INTO AcessoLaboratorios(matriculaAluno, nomeAluno,
      sobrenomeAluno, entrada, login, logout, saida)\nVALUES")
registro_acessos = []
id = 1
mes = 4
ending = ','

for dia in range(1, 24):
    quant_access = random.randrange(5, 15)
    acessos = {}
    if dia == 21 or dia == 22 or dia == 23:
        acessos[152354] = True
        generateCaio(dia)
    for i in range(0, quant_access):
        if dia == 23 and i == quant_access -1:
            ending = ';'
        aluno = alunos[random.randrange(0, len(alunos))]
        matriculaAluno = aluno[0]
        nomeAluno = aluno[1]
        sobrenomeAluno = aluno[2]
        while matriculaAluno in acessos:
            aluno = alunos[random.randrange(0, len(alunos))]
            matriculaAluno = aluno[0]
            nomeAluno = aluno[1]
            sobrenomeAluno = aluno[2]
        acessos[matriculaAluno] = True
        entrada, saida = rand_access(dia, mes, 8)

        delay = random.randrange(5, 10)
        minute = (entrada.minute + delay) % 60

```

```

hour = entrada.hour + int((entrada.minute + delay) / 60)
login = datetime.datetime(entrada.year, entrada.month, entrada.
    day, hour, minute, random.randrange(0, 59))

delay = random.randrange(5, 15)
minute = (saida.minute - delay) % 60
hour = saida.hour + int((saida.minute - delay) / 60)
logout = datetime.datetime(saida.year, saida.month, saida.day,
    hour, minute, random.randrange(0, 59))

print("      ('" + str(matriculaAluno) + "', '" + str(nomeAluno) + "
    "', '" + str(sobrenomeAluno) + "', '" + str(entrada) + "', '"
    + str(login) + "', '" + str(logout) + "', '" + str(saida) + "
    "')" + ending)
registro_acessos.append([id, matriculaAluno, entrada, saida])
id += 1
print("\n")

# Acesso aos Computadores - gera acessos randomicos para os alunos ao
# computador, levando em conta a tabela AcessoLaboratorios

ending = ','

print("INSERT INTO AcessoComputadores(id, matriculaAluno, login, logout)
    \nVALUES")
for i in range(0, len(registro_acessos)):
    registro = registro_acessos[i]
    if i == len(registro_acessos) - 1:
        ending = ';'
    id, matriculaAluno, entrada, saida = registro

    delay = random.randrange(10, 20)
    minute = (entrada.minute + delay) % 60
    hour = entrada.hour + int((entrada.minute + delay) / 60)
    login = datetime.datetime(entrada.year, entrada.month, entrada.day,
        hour, minute, random.randrange(0, 59))

    delay = random.randrange(10, 20)
    minute = (saida.minute + delay) % 60
    hour = saida.hour + int((saida.minute + delay) / 60)
    logout = datetime.datetime(saida.year, saida.month, saida.day, hour,
        minute, random.randrange(0, 59))
    print("      (" + str(id) + ", '" + str(matriculaAluno) + "', '" + str
        (login) + "', '" + str(logout) + "')" + ending)

print("\n")

```

```

# Controle de Seguranças - Preenche a tabela

relatorios = ["Turno correu normalmente.", "Nao houve nenhum incidente."
, "Nenhum incidente a relatar", "Nenhum incidente", "Nada a relatar",
    "Tudo nos conformes", "Nada relevante a relatar"]

def relatorio():
    return relatorios[random.randrange(0, len(relatorios))]

def seguranca_print(hentrada, hsaida):
    seguranças_do_dia = []
    entrada = datetime.datetime(2022, mes, dia, hentrada, random.
        randrange(0, 15), random.randrange(0, 59))
    saida = datetime.datetime(2022, mes, dia, hsaida, random.randrange
        (0, 15), random.randrange(0, 59))
    for i in range(0, 3):
        idSeguranca = random.choice(security_m).id
        while idSeguranca in seguranças_do_dia:
            idSeguranca = random.choice(security_m).id
        seguranças_do_dia.append(idSeguranca)
        print("      ('" + str(idSeguranca) + "', '" + str(entrada) + "',
            '" + str(saida) + "', '" + relatorio() + "')")

print("INSERT INTO ControleSeguranças(idSeguranca, entrada, saida,
relatorio)\nVALUES")
for mes in range(4, 6):
    for dia in range(1, 31):
        if dia == 31 and mes in [4, 6]:
            continue
        seguranca_print(7, 15)
        seguranca_print(13, 21)

```

## D.2 FASE 2

Listing D.2 – Geração SQL Tabelas

```

# Metodos de Geracao

import random
import datetime

nomes = [
    "Andre", "Alice", "Alfredo", "Arthur", "Armando", "Adriana", "
    Adriano", "Alberto",

```

```

"Beatriz", "Bianca", "Breno", "Bruno", "Bruna", "Bob", "Bernardo", "
    Benjamin", "Bento", "Berenice", "Barney", "Barbara",
"Caio", "Caroline", "Carolina", "Carlor", "Cauan", "Cleberson", "
    Cicero", "Cecilia", "Davi", "Camila",
"Eduardo", "Eduarda", "Estevam", "Elena", "Elenilce", "Ester", "
    Everton", "Elen", "Elizandra", "Elizete",
"Fabio", "Fabiana", "Fernando", "Fernanda", "Fatima", "Felipe", "
    Flavia",
"Gabriel", "Gabriela", "Gustavo", "Galvao", "Giovani", "Giovana", "
    Gerson",
"Hugo", "Helena", "Helen",
"Igor", "Iasmim", "Ivan", "Isaac", "Ivone",
"Joao", "Joana", "Jonas", "Julio", "Juliana", "Josue", "Jhonatan", "
    Jardel", "Janaina", "Joyce", "Joseli", "Joselaine", "Jasmine", "
    Juan", "Jade",
"Kerolin", "Kauan",
"Leticia", "Lucas", "Luan", "Lilian", "Laura", "Leonardo",
"Marcos", "Maicon", "Marcelo", "Marcela", "Miriam", "Mauricio", "
    Mario", "Mariana",
"Natan", "Natalia",
"Orlando", "Oscar", "Osmar",
"Pedro", "Pietro", "Paula", "Paulo", "Pamela", "Paloma", "Patricia", "
    Patricio", "Penelope", "Priscila",
"Renato", "Renata", "Rodrigo", "Rubens", "Ricardo", "Roberta",
"Susan", "Savio", "Suelen", "Silmar",
"Tuane", "Thiago", "Tatiana",
"Victor", "Viviane", "Vivian", "Velma", "Valkiria",
"Yasmim"

```

```
]
```

```
sobrenomes = [
```

```

"Sousa", "Souza", "Sant Anna", "Brum", "Goncalves", "Ferreira", "
    Nascimento", "do Nascimento", "Santos", "dos Santos", "de Sant
    Anna", "Ribeiro", "Pereira",
"Outeiro", "Campos", "Matos", "Silva", "da Silva", "Arnaud", "Scholl
", "Rodrigues", "Carvalho", "Luna", "Lima", "de Lima", "de
    Carvalho", "Gomes", "Oliveira",
"Alves", "Martins", "Cerqueira", "Barros", "Barbosa", "Andrade", "
    Batista", "Cardoso", "Castro", "Costa", "Dias", "Duarte", "
    Freitas", "Garcia", "Lopes", "Felix",
"Fernandes", "Machado", "Marques", "Medeiros", "Melo", "Monteiro", "
    Moreira", "Nunes", "Oliveira", "Moura", "Reis", "Rocha", "
    Teixeira", "Vieira"

```

```
]
```

```
nomesEmpresas = [
```

```

"Dope Entregas", "Legendary Entregas", "Mercy Entregas", "Burst
  Entregas", "Prospect Entregas", "Entregasex", "Spin Entregas", "
  Box Entregas", "Spirit Entregas", "Central Entregas", "
  Entregasadora",
"Metro Entregas", "Creative Entregas", "Boizoid", "Boicog", "Boiish"
  , "Boiscape", "Boiporium", "Boinest", "Boiopolis", "Boitastic", "
  Boiiva", "Boiado", "Boiify", "Boique", "Boihut",
"Boister", "Boizen", "Boinetic", "Boiadil", "Boigenics"
]

matriculas_usadas = {'152354': True, '155623': True, '163224': True}

def nova_matricula():
    value = random.randrange(1, 20000)
    while str(value) in matriculas_usadas:
        value = random.randrange(1, 20000)
    matriculas_usadas[str(value)] = True
    return str(value)

def rand_cargo_dev():
    tipos = ['desenvolvedor_junior', 'desenvolvedor_pleno', '
        desenvolvedor_senior', 'especialista', 'dev_ops' ]
    return tipos[random.randrange(0, len(tipos))]

def rand_cargo_atendimento():
    tipos = ['repcionista', 'telefonista']
    return tipos[random.randrange(0, len(tipos))]

def rand_cargo_rh():
    tipos = ['consultor', 'gestor', 'assistente']
    return tipos[random.randrange(0, len(tipos))]

def rand_cargo_juridico():
    tipos = ['advogado', 'assistente', 'estagiario']
    return tipos[random.randrange(0, len(tipos))]

def rand_cargo_logistica():
    tipos = ['entregador', 'coordenador', 'estoquista', 'motorista']
    return tipos[random.randrange(0, len(tipos))]

def rand_cargo_gerencia():
    tipos = ['gerente_de_projeto', 'gerente_de_area']
    return tipos[random.randrange(0, len(tipos))]

def rand_area():
    tipos = ['desenvolvimento', 'Atendimento_ao_cliente', 'RH', '
        juridico', 'logistica', 'gerencia']

```

```

return tipos[random.randrange(0, len(tipos))]

def rand_telephone():
    ddds = [11, 12, 13, 14, 15, 16, 17, 18, 19, 21, 22,
            24, 27, 28, 31, 32, 33, 34,
            35, 37, 38, 41, 42, 43, 44, 45, 46,
            47, 48, 49, 51, 53, 54, 55,
            61, 62,
            63, 64, 65, 66, 67, 68, 69, 71, 73,
            74, 75, 77, 79, 81, 82, 83,
            84, 85,
            86, 87, 88, 89, 91, 92, 93, 94,
            95, 96, 97, 98, 99]
    return "(" + str(dds[random.randrange(0, len(dds))]) + ") " + str(
        random.randrange(1000, 9999)) + "-" + str(random.randrange(1000,
        9999))

def rand_data_nascimento():
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11",
               ", "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10",
               ", "11", "12"]
    dia = random.randrange(1, 31)

    if dia == 31:
        mes = meses31[random.randrange(0, len(meses31))]
    elif dia > 28:
        mes = meses30[random.randrange(0, len(meses30))]
    else:
        mes = meses28[random.randrange(0, len(meses28))]
    ano = random.randrange(1994, 2003)
    return str(ano) + "-" + mes + "-" + str(dia)

def rand_data_nascimento():
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11",
               ", "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10",
               ", "11", "12"]
    dia = random.randrange(1, 31)

    if dia == 31:
        mes = meses31[random.randrange(0, len(meses31))]
    elif dia > 28:
        mes = meses30[random.randrange(0, len(meses30))]
    else:

```

```

        mes = meses28[random.randrange(0, len(meses28))]
        ano = random.randrange(1964, 2000)
        return str(ano) + "-" + mes + "-" + str(dia)

def rand_gasto_e_receita_data(mes):
    dia = random.randrange(1, 30)
    ano = random.randrange(2022)
    return str(2022) + "-" + str(mes) + "-" + str(dia)

limite_valor_individual = 1000000
def rand_gasto():
    return random.randrange(1000, limite_valor_individual)

def rand_destino_gasto():
    destinos = ["conta de luz", "conta de agua", "conta de internet", "
        conta de telefone", "gastos de viagem", "almoco de funcionario",
        "fornecedores", "coffe break", "aluguel", "despesas de veiculo",
        "outros"]
    return destinos[random.randrange(0, len(destinos))]

def rand_empresa():
    return nomesEmpresas[random.randrange(0, len(nomesEmpresas))]

# Dados de Funcionarios

funcionarios = []
print("INSERT INTO Funcionarios(matricula, nome, sobrenome, telefone,
    dataDeNascimento, cargo, area, gerente, pode_mover_dinheiro,
    pode_acessar_recursos_restritos, pode_desligar_funcionarios) \nVALUES
    ")
print("    ('22354', 'Paulo', 'Pinho', '" + rand_telephone() + "', '" +
    rand_data_nascimento() + "', 'supervisor', 'financeiro', 'Marcos
    Antunes', true, true, false),")
print("    ('21899', 'Pedro', 'Lima', '" + rand_telephone() + "', '" +
    rand_data_nascimento() + "', 'supervisor', 'financeiro', 'Lilian
    Neiva', true, true, false),")
print("    ('26548', 'Lucas', 'Souza', '" + rand_telephone() + "', '" +
    rand_data_nascimento() + "', 'analista', 'financeiro', 'Maria
    Angelica', true, false, false),")
print("    ('34954', 'Marta', 'Lobo', '" + rand_telephone() + "', '" +
    rand_data_nascimento() + "', 'auxiliar', 'financeiro', 'Marcos
    Antunes', true, false, false),")
print("    ('27845', 'Joaquina', 'Osorio', '" + rand_telephone() + "', '
    " + rand_data_nascimento() + "', 'analista', 'financeiro', 'Marcos
    Antunes', true, false, false),")
print("    ('39578', 'Matheus', 'Castilho', '" + rand_telephone() + "',
    '" + rand_data_nascimento() + "', 'auxiliar', 'financeiro', 'Marcos

```

```
Antunes', true, false, false),")
print("      ('33654', 'Lucas', 'Carreiro', '"' + rand_telephone() + "', '"
      + rand_data_nascimento() + "', 'supervisor', 'financeiro', 'Marcos
      Antunes', true, true, false),")
print("      ('22222', 'Paulo', 'Seixas', '"' + rand_telephone() + "', '" +
      rand_data_nascimento() + "', 'analista', 'financeiro', 'Marcos
      Antunes', true, false, false),")

count = 0
total = 300

for i in range(0, total):
    nomePos = random.randrange(0, len(nomes))
    sobrenomePos = random.randrange(0, len(sobrenomes))
    nomeGerentePos = random.randrange(0, len(nomes))
    sobrenomeGerentePos = random.randrange(0, len(sobrenomes))
    nomeGerente = nomes[nomeGerentePos] + " " + nomes[
        sobrenomeGerentePos]

    area = rand_area()
    pode_desligar_funcionarios = ''
    cargo = ''
    if area == 'desenvolvimento':
        cargo = rand_cargo_rh()
        pode_desligar_funcionarios = 'FALSE'
    elif area == 'Atendimento_ao_cliente':
        cargo = rand_cargo_dev()
        pode_desligar_funcionarios = 'FALSE'
    elif area == 'RH':
        cargo = rand_cargo_rh()
        pode_desligar_funcionarios = 'TRUE'
    elif area == 'juridico':
        cargo = rand_cargo_juridico()
        pode_desligar_funcionarios = 'FALSE'
    elif area == 'logistica':
        cargo = rand_cargo_logistica()
        pode_desligar_funcionarios = 'FALSE'
    elif area == 'gerencia':
        cargo = rand_cargo_gerencia()
        pode_desligar_funcionarios = 'FALSE'

    matricula = nova_matricula();
    nome = nomes[nomePos]
    sobrenome = sobrenomes[sobrenomePos]

    funcionario = {"matricula": matricula, "nome": nome, "sobrenome" :
        sobrenome}
```

```

funcionarios.append(funcionario)

ending = ','
if count == total - 1:
    ending = ';'

print("      ('" + matricula + "', '" + nome + "', '" + sobrenome + "',
      '" + rand_telephone() + "', '" + rand_data_nascimento() + "', '"
      + cargo + "', '" + area + "', '" + nomeGerente + "', FALSE,
      FALSE, " + pode_desligar_funcionarios + ")" + ending )
count += 1

# Gastos

print("INSERT INTO Gastos(id, data, mes_referencia,
      matricula_responsavel, nome_responsavel, sobrenome_responsavel, valor
      , destino) \nVALUES")

paulo = {3: "      (3, '2022-5-21', '5', 22354, 'Paulo', 'Pinho', 380109,
      'Outros'),",
      45: "      (45, '2022-5-15', '5', 22354, 'Paulo', 'Pinho', 292687,
      'Outros'),",
      78: "      (78, '2022-5-2', '5', 22354, 'Paulo', 'Pinho', 10934,
      'Outros'),",
      129: "      (129, '2022-5-6', '5', 22354, 'Paulo', 'Pinho',
      61144, 'Outros'),",
      131: "      (131, '2022-5-22', '5', 22354, 'Paulo', 'Pinho',
      87793, 'Outros'),",
      148: "      (148, '2022-5-29', '5', 22354, 'Paulo', 'Pinho',
      52670, 'Outros'),",
      190: "      (190, '2022-5-30', '5', 22354, 'Paulo', 'Pinho',
      56821, 'Outros'),",
      231: "      (231, '2022-5-20', '5', 22354, 'Paulo', 'Pinho',
      177996, 'Outros'),",
      232: "      (232, '2022-5-4', '5', 22354, 'Paulo', 'Pinho',
      36437, 'Outros'),",
      290: "      (290, '2022-5-17', '5', 22354, 'Paulo', 'Pinho',
      38279, 'Outros'),",
      343: "      (343, '2022-5-9', '5', 22354, 'Paulo', 'Pinho',
      48196, 'Outros'),",
      347: "      (347, '2022-5-1', '5', 22354, 'Paulo', 'Pinho',
      56934, 'Outros'),"}

pedro = {7: "      (7, '2022-5-21', '5', 21899, 'Pedro', 'Lima', 211280, '
      conta de luz'),",
      32: "      (32, '2022-5-15', '5', 21899, 'Pedro', 'Lima', 160727,
      'conta de telefone'),",

```

```

98: "      (98, '2022-5-2', '5', 21899, 'Pedro', 'Lima', 240912,
      'conta de luz'),",
278: "      (278, '2022-5-6', '5', 21899, 'Pedro', 'Lima',
      495041, 'almoco'),",
279: "      (279, '2022-5-22', '5', 21899, 'Pedro', 'Lima',
      59141, 'fornecedores'),",
352: "      (352, '2022-5-29', '5', 21899, 'Pedro', 'Lima',
      60583, 'gastos de viagens'),",
399: "      (399, '2022-5-30', '5', 21899, 'Pedro', 'Lima',
      72316, 'conta de agua'),"}

lucas = {46: "      (46, '2022-5-21', '5', 26548, 'Lucas', 'Souza',
      650000, 'Fornecedores'),",
      52: "      (52, '2022-5-15', '5', 26548, 'Lucas', 'Souza', 650000,
      'Fornecedores'),"}

count = 1

limiteTransacoes = 5
transacoes = {}
limite_valor_gasto = 210345948
restante = limite_valor_gasto

for i in range(5,7):

    mes = i
    if mes == 5:
        restante = limite_valor_gasto
    else:
        restante = 124578435
    while restante > 0:

        printou = False
        for id in paulo:
            if count == id:
                print(paulo[id])
                printou = True
                break
        for id in pedro:
            if count == id:
                print(pedro[id])
                printou = True
                break
        for id in lucas:
            if count == id:
                print(lucas[id])
                printou = True
                break

```

```

if printou:
    count += 1
    continue

data = rand_gasto_e_receita_data(mes)

funcionario = funcionarios[random.randrange(0, len(funcionarios)
)]
if funcionario["nome"] in transacoes:
    while transacoes[funcionario["nome"]] > limiteTransacoes :
        funcionario = funcionarios[random.randrange(0, len(
            funcionarios))]
        transacoes[funcionario["nome"]] += 1

valor = rand_gasto()
if valor > restante:
    valor = restante
    restante = 0
    if mes == 6:
        ending = ';'
    else:
        ending = ','
else:
    restante -= valor
    ending = ','

destino = rand_destino_gasto()
print("      (" + str(count) + ", ' " + data + "', ' " + str(mes) + "
', " + str(funcionario["matricula"]) + ", ' " + funcionario["
nome"] + "', ' " + funcionario["sobrenome"] + "', " + str(
valor) + ", ' " + destino + "') " + ending)
count += 1

# Receita

print("INSERT INTO Receita(id, data, mes_referencia, responsavel, valor)
\nVALUES")

count = 0

limiteTransacoes = 5
transacoes = {}
limite_valor_receita = 210345948
restante = limite_valor_receita

for i in range(5,7):

```

```

mes = i
if mes == 5:
    restante = limite_valor_gasto
else:
    restante = 124578435
while restante > 0:
    data = rand_gasto_e_receita_data(mes)
    empresa = rand_empresa()

    valor = rand_gasto()
    if valor > restante:
        valor = restante
        restante = 0
        if mes == 6:
            ending = ','
        else:
            ending = ','
    else:
        restante -= valor
        ending = ','

    destino = rand_destino_gasto()
    print("      (" + str(count) + ", '" + data + "', '" + str(mes) + "
          ', '" + empresa + "', " + str(valor) + ")" + ending)
    count += 1

# Dados anuais da empresa

print("INSERT INTO Dados_da_empresa(Nome, Ano,
      Quantidade_de_funcionarios, Receita_anual, Transacoes_por_funcionario
      )\nVALUES")

quantFunc = 6248
receita = 1000000

for ano in range (2010, 2023):
    if ano < 2015:
        nome = "Boizao"
    else:
        nome = "Boi Livre"
    quantFunc += 6973
    if ano < 2016:
        receita += 856945
    else:
        receita += 3649527
    if ano < 2018:
        tran = 20

```

```

else:
    tran = 10

print("      ('" + nome + "', '" + str(ano) + "', '" + str(quantFunc)
      + "', '" + str(receita) + "', '" + str(tran) + ''),")

```

### D.3 FASE 3

#### Listing D.3 – Geração SQL Tabelas

```

import random
import datetime

# Metodos de Geracao

nomes = [
    "Andre", "Alice", "Alfredo", "Arthur", "Armando", "Adriana", "
    Adriano", "Alberto",
    "Beatriz", "Bianca", "Breno", "Bruno", "Bruna", "Bob", "Bernardo", "
    Benjamin", "Bento", "Berenice", "Barney", "Barbara",
    "Caio", "Caroline", "Carolina", "Carlora", "Cauan", "Cleberson", "
    Cicero", "Cecilia", "Davi", "Camila",
    "Eduardo", "Eduarda", "Estevam", "Elena", "Elenilce", "Ester", "
    Everton", "Elen", "Elizandra", "Elizete",
    "Fabio", "Fabiana", "Fernando", "Fernanda", "Fatima", "Felipe", "
    Flavia",
    "Gabriel", "Gabriela", "Gustavo", "Galvao", "Giovani", "Giovana", "
    Gerson",
    "Hugo", "Helena", "Helen",
    "Igor", "Iasmim", "Ivan", "Isaac", "Ivone",
    "Joao", "Joana", "Jonas", "Julio", "Juliana", "Josue", "Jhonatan", "
    Jardel", "Janaina", "Joyce", "Joseli", "Joselaine", "Jasmine", "
    Juan", "Jade",
    "Kerolin", "Kauan",
    "Leticia", "Lucas", "Luan", "Lilian", "Laura", "Leonardo",
    "Marcos", "Maicon", "Marcelo", "Marcela", "Miriam", "Mauricio", "
    Mario", "Mariana",
    "Natan", "Natalia",
    "Orlando", "Oscar", "Osmar",
    "Pedro", "Pietro", "Paula", "Paulo", "Pamela", "Paloma", "Patricia",
    "Patricio", "Penelope", "Priscila",
    "Renato", "Renata", "Rodrigo", "Rubens", "Ricardo", "Roberta",
    "Susan", "Savio", "Suelen", "Silmar",
    "Tuane", "Thiago", "Tatiana",
    "Victor", "Viviane", "Vivian", "Velma", "Valkiria",

```

```

    "Yasmim"
]

sobrenomes = [
    "Sousa", "Souza", "Sant Anna", "Brum", "Goncalves", "Ferreira", "
    Nascimento", "do Nascimento", "Santos", "dos Santos", "de Sant
    Anna", "Ribeiro", "Pereira",
    "Outeiro", "Campos", "Matos", "Silva", "da Silva", "Arnaud", "Scholl
    ", "Rodrigues", "Carvalho", "Luna", "Lima", "de Lima", "de
    Carvalho", "Gomes", "Oliveira",
    "Alves", "Martins", "Cerqueira", "Barros", "Barbosa", "Andrade", "
    Batista", "Cardoso", "Castro", "Costa", "Dias", "Duarte", "
    Freitas", "Garcia", "Lopes", "Felix",
    "Fernandes", "Machado", "Marques", "Medeiros", "Melo", "Monteiro", "
    Moreira", "Nunes", "Oliveira", "Moura", "Reis", "Rocha", "
    Teixeira", "Vieira"
]

nomesEmpresas = [
    "Dope Entregas", "Legendary Entregas", "Mercy Entregas", "Burst
    Entregas", "Prospect Entregas", "Entregasex", "Spin Entregas", "
    Box Entregas", "Spirit Entregas", "Central Entregas", "
    Entregasadora",
    "Metro Entregas", "Creative Entregas", "Boizoid", "Boicog", "Boiish"
    , "Boiscape", "Boioporium", "Boinest", "Boiopolis", "Boitastic", "
    Boiiva", "Boiado", "Boiify", "Boique", "Boihut",
    "Boister", "Boizen", "Boinetic", "Boiadil", "Boigenics"
]

matriculas_usadas = {'152354': True, '155623': True, '163224': True}

def nova_matricula():
    value = random.randrange(1, 20000)
    while str(value) in matriculas_usadas:
        value = random.randrange(1, 20000)
    matriculas_usadas[str(value)] = True
    return str(value)

def rand_data_inicial():
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11
    ", "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10
    ", "11", "12"]
    dia = random.randrange(1, 31)

    if dia == 31:

```

```

        mes = random.choice(meses31)
    elif dia > 28:
        mes = random.choice(meses30)
    else:
        mes = random.choice(meses28)
    ano = random.randrange(1994, 2012)
    return str(ano) + "-" + mes + "-" + str(dia), dia, mes, ano

def rand_data_final(diaInicial, mesInicial, anoInicial):
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11",
               "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10",
               "11", "12"]

    while mesInicial > meses31[0]:
        meses31.pop(0)
    while mesInicial > meses30[0]:
        meses30.pop(0)
    while mesInicial > meses28[0]:
        meses28.pop(0)

    dia = random.randrange(diaInicial, 31)

    if dia == 31:
        mes = random.choice(meses31)
    elif dia > 28:
        mes = random.choice(meses30)
    else:
        mes = random.choice(meses28)
    ano = random.randrange(anoInicial, 2022)

    return str(ano) + "-" + mes + "-" + str(dia)

def rand_data_intervalo(diaInicial, mesInicial, anoInicial, diaFinal,
mesFinal, anoFinal):
    if anoInicial == anoFinal:
        ano = anoInicial
        if mesInicial == mesFinal:
            mes = mesInicial
            dia = random.randrange(diaInicial, diaFinal)
        else:
            mes = random.randrange(mesInicial, mesFinal+1)
            if mes == mesInicial:
                dia = random.randrange(diaInicial, 28)
            elif mes == mesFinal:
                dia = random.randrange(1, diaFinal)

```

```

        else:
            dia = random.randrange(1, 28)

    else:
        ano = random.randrange(anoInicial, anoFinal+1)
        if ano == anoInicial:
            mes = random.randrange(mesInicial, 13)
        elif ano == anoFinal:
            mes = random.randrange(1, mesFinal)
        else:
            mes = random.randrange(1, 13)

    if mes == mesInicial:
        dia = random.randrange(diaInicial, 28)
    elif mes == mesFinal:
        dia = random.randrange(1, diaFinal)
    else:
        dia = random.randrange(1, 28)

    if mes < 10:
        mes = "0" + str(mes)

    return str(ano) + "-" + str(mes) + "-" + str(dia)

def rand_motivo_desligamento():
    motivos = ['baixo rendimento', 'perfil nao se enquadra', 'justa
        causa', 'saida voluntaria', 'abandono', 'violacao de conduta', '
        negligencia na funcao']
    return random.choice(motivos)

def rand_nivel():
    nivel = ['leve', 'medio', 'grave']
    return random.choice(nivel)

def rand_incidente_leve():
    incidente = ['Alguma equipe tomou todo o cafe e nao o repos', '
        Alguem sujou completamente o banheiro e desapareceu', 'Haviam
        pessoas conversando extremamente alto no corredor']
    return random.choice(incidente)

def rand_incidente_medio():
    nomeAlvo = random.choice(nomes) + " " + random.choice(sobrenomes)
    nomeAgente = random.choice(nomes) + " " + random.choice(sobrenomes)

    incidente = ['O(A) funcionario(a) ' + nomeAgente + ' estava gritando
        com o(a) fucionario(a) ' + nomeAlvo, 'O(A) funcionario(a) ' +
        nomeAgente + ' estava fazendo bullying com o(a) fucionario(a) ' +

```

```

        nomeAlvo]
    return random.choice(incidente)

def rand_incidente_grave():
    nomeAlvo = random.choice(nomes) + " " + random.choice(sobrenomes)

    funcionario = random.choice(funcionarios)
    nomeAgente = funcionario[1] + " " + funcionario[2]

    incidente = ['O(A) funcionario(a) ' + nomeAgente + ' agrediu
        fisicamente o(a) funcionario(a) ' + nomeAlvo, 'O(A) funcionario(a)
        ' + nomeAgente + ' assediou o(a) funcionario(a) ' + nomeAlvo]
    return random.choice(incidente)

def rand_telephone():
    return "(21) 9" + str(random.randrange(6, 9)) + str(random.randrange
        (100, 999)) + "-" + str(random.randrange(1000, 9999))

def rand_mensagem_telemarketing():
    mensagens = ['Ola, voce acaba de ganhar 1000GB de internet, e so
        responder 1 para ativar isso tudo', 'Temos uma promocao
        imperdivel pra voce, ligue para (11)11111111 para saber mais',
        'Que tal turbinar a sua internet, responda com EU QUERO
        para ativar', 'Somos do do grupo da libertacao liberta,
        e queremos sua ajuda, ligue para (22)22222222 para fazer
        sua doacao']
    return random.choice(mensagens)

def rand_mensagem_tinder():
    mensagens = ['Tem um barzinho aqui pert de casa, quer vir?', 'Que
        tal irmos no shopping amanha?',
        'Vamos num cineminha, adoro o escurinho', 'Fiquei sabendo
        de uma balada nova na cidade, ta afim de ir?']
    return random.choice(mensagens)

# Dados de Desligamento

funcionarios = [
    ('34895', 'Felicia', 'Castro'),
    ('26849', 'Marcos', 'Costa'),
    ('30012', 'Lucas', 'Silveira')
]

print("INSERT INTO Desligamentos(matricula, nome, sobrenome,
    data_entrada, data_saida, motivo_desligamento) \nVALUES")

```

```

print("      ('34895', 'Felicia', 'Castro', '2015-06-29', '2022-02-15', '
    nao declarado'),")
print("      ('26849', 'Marcos', 'Costa', '2003-03-17', '2022-02-17', '
    perfil nao se enquadra'),")
print("      ('30012', 'Lucas', 'Silveira', '2019-09-08', '2022-02-16', '
    baixo rendimento'),")
print("      ('24877', 'Fabio', 'Campos', '2020-09-08', '2022-02-10', '
    agressao'),")

count = 0
total = 100

for i in range(0, total):
    motivo = rand_motivo_desligamento()

    matricula = nova_matricula()
    nome = random.choice(nomes)
    sobrenome = random.choice(sobrenomes)

    funcionarios.append((matricula, nome, sobrenome))

    dataInicial, diaInicial, mesInicial, anoInicial = rand_data_inicial
        ()
    dataFinal = rand_data_final(diaInicial, mesInicial, anoInicial)

    ending = ','
    if count == total - 1:
        ending = ';'

    print("      ('"+ matricula + "', '" + nome + "', '" + sobrenome + "',
        '" + dataInicial + "', '" + dataFinal + "', '" + motivo + "')" +
        ending )
    count += 1

# Incidentes

print("INSERT INTO Incidentes(id, nome, sobrenome, nivel, reclamacao) \
    nVALUES")
incidentes = {2: "      ('2', 'Felicia', 'Castro', 'grave', '0 funcionario
    Caio Porto proferiu falas machistas para a funcionaria Felicia
    Castro, falando que a mesma devia ir lavar uma louca em vez de
    reclamar de salario'),"}
incidentes[6] = "      ('6', 'Felicia', 'Castro', 'grave', '0 funcionario
    Caio Porto pediu para ver os seios da funcionaria Felicia Castro'),"
incidentes[8] = "      ('8', 'Fabio', 'Campos', 'leve', 'Roubaram o bife
    da marmita de Fabio Campos, o mesmo diz achar que foi Davi Duarte'),"

```

```

incidentes[12] = "      ('12', 'Davi', 'Duarte', 'leve', 'Davi Duarte
      alega que Fabio Campos instalou uma extensao do Nicholas Cage em seu
      navegador, o mesmo diz ser retaliacao pelo seu bife roubado'),"
incidentes[13] = "      ('13', 'Felicia', 'Castro', 'grave', 'O
      funcionario Caio Porto trancou a fucionaria Felicia Castro em uma
      sala dizendo que so a deixaria sair quando ela tirasse a roupa, o
      incidente nao foi mais longe devido a chegada de colegas de Felicia')
      ,"
incidentes[15] = "      ('15', 'Fabio', 'Campos', 'leve', 'Novamente
      roubaram o bife da marmita de Fabio Campos'),"
incidentes[16] = "      ('16', 'Davi', 'Duarte', 'medio', 'Fabio Campos
      gritou com Davi Duarte por achar que o mesmo roubou seu bife'),"
incidentes[22] = "      ('22', 'Fabio', 'Campos', 'leve', 'Novamente
      roubaram o bife da marmita de Fabio Campos, o mesmo diz que seu pote
      estava completamente revirado e diz que Davi Duarte estava rindo dele
      '),"
incidentes[25] = "      ('25', 'Davi', 'Duarte', 'grave', 'Davi Duarte
      alega que Fabio Campos o atacou e quebrou dois dentes de Davi apos o
      mesmo ter rido de Fabio ter se irritado com a geladeira por algum
      motivo'),"
incidentes[27] = "      ('27', 'Felicia', 'Castro', 'grave', 'O
      funcionario Caio Porto tentou beijar a funcinaria Felicia sem sua
      permissao'),"
incidentes[28] = "      ('28', NULL, NULL, 'leve', ' O funcionario Lucas
      Silveira passa grande parte do tempo na maquina de cafe sem trabalhar
      '),"

count = 0
total = 40

for i in range(0, total):
    if i in incidentes:
        print(incidentes[i])
        count += 1
        continue

    nivel = rand_nivel()

    if nivel == 'leve':
        reclamacao = rand_incidente_leve()
    elif nivel == 'medio':
        reclamacao = rand_incidente_medio()
    elif nivel == 'grave':
        reclamacao = rand_incidente_grave()

    ending = ','

```

```

if count == total - 1:
    ending = ';'

if random.randrange(0, 2):
    funcionario = random.choice(funcionarios)
    nome = funcionario[1]
    sobrenome = funcionario[2]
    print("      ('"+ str(i) + "', '" + nome + "', '" + sobrenome + "
        ', '" + nivel + "', '" + reclamacao + "')" + ending )
else:
    print("      ('"+ str(i) + "', NULL, NULL, '" + nivel + "', '" +
        reclamacao + "')" + ending )

count += 1

# Mensagens

def check_mensagem(i):
    while True:
        if i in mensagens:
            print(mensagens[i])
            i += 1
            continue
        else:
            break
    return i

telefoneCaio = "(21)987654321"
dia2015 = "'2015-07-02'"
dia13 = "'2022-05-13'"
dia15 = "'2022-05-15'"
dia16 = "'2022-05-16'"
dia18 = "'2022-05-18'"
dia20 = "'2022-05-20'"

print("INSERT INTO Mensagens(id, telefone_de_origem, telefone_de_destino
    , nome_origem, nome_destino, data, mensagem) \nVALUES")
mensagens = {23: "      ('23', '(21)123456789', '(21)987654321', 'Felicia
    Castro', 'Caio Porto', " + dia2015 + " , 'Oi Caio, tudo bem?')"}
mensagens[24] = "      ('24', '(21)987654321', '(21)123456789', 'Caio
    Porto', 'Felicia Castro', " + dia2015 + " , 'Oi quem e voce?')",
mensagens[25] = "      ('25', '(21)123456789', '(21)987654321', 'Felicia
    Castro', 'Caio Porto', " + dia2015 + " , 'Sou a Felicia, la da
    empresa, entrei esse dias, espero que a gente se de bem'),"
mensagens[26] = "      ('26', '(21)987654321', '(21)123456789', 'Caio
    Porto', 'Felicia Castro', " + dia2015 + " , 'Ah prazer Felicia,

```

```
espero que a gente se de muito bem mesmo :)')),"

mensagens[105] = " ('105', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', '2020-02-10', 'Oi Felicia, tava pensando
aqui, que tal sairmos pra beber alguma coisa amanha?')),"
mensagens[106] = " ('106', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', '2020-02-10', 'Caio, ja falei pra voce, nao
quero sair com vc, para de me encher o saco')),"

mensagens[180] = " ('180', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia13 + " , 'Boa tarde senhor Caio, tudo
bem?')),"
mensagens[181] = " ('181', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia13 + " , 'Oi Lucas, fala ai')),"
mensagens[182] = " ('182', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia13 + " , 'Senhor Caio, gostaria de
tirar uma duvida como o senhor sobre o relatorio de ontem')),"
mensagens[183] = " ('183', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia13 + " , 'O senhor poderia vir na
minha mesa?')),"
mensagens[184] = " ('184', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia13 + " , 'Ja sai hoje Lucas, amanha
nos vemos')),"

mensagens[205] = " ('205', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Boa noite senhor Caio, tudo
bem?')),"
mensagens[206] = " ('206', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Acho que senhor se esqueceu
de me ajudar com aquele relatorio')),"
mensagens[207] = " ('207', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Preciso entrega-lo hoje
ainda, voce poderia me ajudar aqui?')),"
mensagens[208] = " ('208', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia15 + " , 'Lucas, voce devia ter me
lembrado antes, eu ja sai do escritorio')),"
mensagens[209] = " ('209', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia15 + " , 'Faca o que voce achar
certo')),"

mensagens[210] = " ('210', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia16 + " , 'Oi Felicia, fiquei sabendo
do que aconteceu com vc, que pena nao e mesmo')),"
mensagens[211] = " ('211', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia16 + " , 'Para de falsidade Caio, eu
sei que foi voce que pediu minha demissao')),"
```

```
mensagens [212] = "      ('212', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro', " + dia16 + " , 'Nao linda, que isso,
      jamais perderia a chance de poder ver vc todos os dias'),"
mensagens [213] = "      ('213', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia16 + " , 'Ah, vai se fuder Caio'),"

mensagens [214] = "      ('214', '(21)976431852', '(21)987654321', 'Marcos
      Costa', 'Caio Porto', " + dia16 + " , 'Oi Caio'),"
mensagens [215] = "      ('215', '(21)976431852', '(21)987654321', 'Marcos
      Costa', 'Caio Porto', " + dia16 + " , 'Tudo bem?'),"
mensagens [216] = "      ('216', '(21)987654321', '(21)976431852', 'Caio
      Porto', 'Marcos Costa', " + dia16 + " , 'Fala Marcos, tudo tranquilo
      e ai?'),"
mensagens [217] = "      ('217', '(21)976431852', '(21)987654321', 'Marcos
      Costa', 'Caio Porto', " + dia16 + " , 'To bem cara, queria saber se a
      Vivian ainda vai demorar'),"
mensagens [218] = "      ('218', '(21)976431852', '(21)987654321', 'Marcos
      Costa', 'Caio Porto', " + dia16 + " , 'Sabe me dizer?'),"
mensagens [219] = "      ('219', '(21)987654321', '(21)976431852', 'Caio
      Porto', 'Marcos Costa', " + dia16 + " , 'Perai, vou procurar ela pra
      saber'),"
mensagens [220] = "      ('220', '(21)987654321', '(21)976431852', 'Caio
      Porto', 'Marcos Costa', " + dia16 + " , 'Cara, ela disse que deve
      estar saindo em mais ou menos 30 minutos'),"
mensagens [221] = "      ('221', '(21)976431852', '(21)987654321', 'Marcos
      Costa', 'Caio Porto', " + dia16 + " , 'Blz, brigadao cara'),"
mensagens [222] = "      ('222', '(21)987654321', '(21)976431852', 'Caio
      Porto', 'Marcos Costa', " + dia16 + " , 'Nada imagina'),"

mensagens [250] = "      ('250', '(21)951753486', '(21)987654321', 'Lucas
      Silveira', 'Caio Porto', " + dia16 + " , 'Caio, bem que todo mundo
      fala que voce e um babaca mesmo'),"
mensagens [251] = "      ('251', '(21)951753486', '(21)987654321', 'Lucas
      Silveira', 'Caio Porto', " + dia16 + " , 'Te pedi ajuda, vc se
      esqueceu de mim, e ainda me demite por incompetencia'),"
mensagens [252] = "      ('252', '(21)951753486', '(21)987654321', 'Lucas
      Silveira', 'Caio Porto', " + dia16 + " , 'Vai se fuder ein'),"
mensagens [253] = "      ('253', '(21)987654321', '(21)951753486', 'Caio
      Porto', 'Lucas Silveira', " + dia16 + " , 'Ue e e culpa minha voce
      nao saber fazer o seu trabalho?'),"
mensagens [254] = "      ('254', '(21)951753486', '(21)987654321', 'Lucas
      Silveira', 'Caio Porto', " + dia16 + " , 'Vai tomar no cu'),"
mensagens [255] = "      ('255', '(21)951753486', '(21)987654321', 'Lucas
      Silveira', 'Caio Porto', " + dia16 + " , 'O usuario Lucas Silverira
      bloqueou Caio porto'),"
mensagens [256] = "      ('256', '(21)987654321', '(21)951753486', 'Caio
      Porto', 'Lucas Silveira', " + dia16 + " , 'Otario'),"
```

```
mensagens [262] = " ('262', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'KRL CAIO, e SeRIO ISSO?! A
VIVAN e MINHA ESPOSA CARA'),"
mensagens [263] = " ('263', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'Como voce pode fazer isso cara
?!'),"
mensagens [264] = " ('264', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'Krl, vc ta fudido'),"
mensagens [265] = " ('265', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'ta muito fudido'),"
mensagens [266] = " ('266', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia18 + " , 'O usuario Caio Porto
bloqueou Marcos Costa'),"

mensagens [267] = " ('267', '(21)123456789', '(21)964875938', 'Felicia
Castro', 'Launa Pascoa', " + dia18 + " , 'Lu, acabei de chegar em
casa, nao to achando meu celular por nada, enem to conseguindo
deslogar do app'),"
mensagens [268] = " ('268', '(21)964875938', '(21)123456789', 'Launa
Pascoa', 'Felicia Castro', " + dia18 + " , 'Caramba amiga, e agora
??'),"
mensagens [269] = " ('269', '(21)123456789', '(21)964875938', 'Felicia
Castro', 'Launa Pascoa', " + dia18 + " , 'Agora e torcer pra quem
tiver com o celular nao fazer nada'),"

mensagens [270] = " ('270', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Sabe Caio, andei pensando
aqui'),"
mensagens [271] = " ('271', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Agora que to sem nada pra
fazer'),"
mensagens [272] = " ('272', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'O que vc acha de sairmos pra
ir no shopping na sexta a noite?'),"
mensagens [273] = " ('273', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia18 + " , 'O que??!! e serio isso?'),"
"
mensagens [274] = " ('274', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Claro que e serio ue, nao
tenho nada pra fazer mesmo, talvez possamos nos conhecer melhor'),"
mensagens [275] = " ('275', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia18 + " , 'Ta bom entao, sexta quando
eu sair do trampo pode ser? umas 20h?'),"
mensagens [276] = " ('276', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Marcado entao'),"
```

```
mensagens[284] = "      ('284', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'Oi Caio ja to aqui te
      esperando com uma surpresinha :p'),"
mensagens[285] = "      ('285', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro'," + dia20 + " , 'Oi Linda, ja to chegando,
      so mais uns 10 minutinhos'),"

mensagens[291] = "      ('291', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro', " + dia20 + " , 'Cheguei, onde vce ta?'),"
mensagens[292] = "      ('292', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'To no banheiro do bloco 13')
      ,"
mensagens[293] = "      ('293', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro', " + dia20 + " , 'Bloco 13?'),"
mensagens[294] = "      ('294', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro', " + dia20 + " , 'Mas o bloco 13 nao ta
      fechado pra obra?'),"
mensagens[295] = "      ('295', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'Eu disse que tinha uma
      surpresinha pra vc nao disse?'),"
mensagens[296] = "      ('296', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'Acho que voce vai querer
      essa surpresa :3'),"
mensagens[297] = "      ('297', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro'," + dia20 + " , 'Epaaa, hoje tem!!'),"

mensagens[298] = "      ('298', '(21)259748631', '(21)987654321', 'Mamae',
      'Caio Porto', " + dia20 + " , 'Cacaoio, pq vc nao esta me atendendo
      ?'),"
mensagens[299] = "      ('299', '(21)259748631', '(21)987654321', 'Mamae',
      'Caio Porto', '2022-05-21' , 'Cacaoio? to preocupada'),"

count = 0
total = 299
i = 0
while i < total:
    i = check_mensagem(i)

    if i < 23:
        data = rand_data_intervalo(1,1,2015,2,7,2015)
    elif i < 105:
        data = rand_data_intervalo(2,7,2015,10,2,2020)
    elif i < 180:
        data = rand_data_intervalo(10,2,2020,13,5,2022)
    elif i < 204:
        data = rand_data_intervalo(13,5,2022,15,5,2022)
    elif i < 250:
```

```

        data = rand_data_intervalo(15,5,2022,16,5,2022)
elif i < 262:
        data = rand_data_intervalo(16,5,2022,18,5,2022)
elif i < 284:
        data = rand_data_intervalo(18,5,2022,20,5,2022)
elif i < 291:
        data = "2022-05-20"

telefone = rand_telephone()

ending = ','
if count == total - 1:
    ending = ';'

# Telemarketing
if random.randrange(0, 3) == 0:
    mensagem = rand_mensagem_telemarketing()
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
          telefoneCaio + "', 'Telemarketing', 'Caio Porto', '" + data +
          "', '" + mensagem + "')" + ending )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefoneCaio + "', '" +
          telefone + "', 'Caio Porto', 'Telemarketing', '" + data + "',
          '0 usuario Caio Porto bloqueou Telemarketing')" + ending )

#Tinder
elif random.randrange(0, 2):
    nome = random.choice(nomes) + " " + random.choice(sobrenomes)
    print("    ('"+ str(i) + "', '" + telefoneCaio + "', '" +
          telefone + "', 'Caio Porto', '" + nome + "', '" + data + "',
          'Oi, tudo bem com vc pessoa linda')" + ending )
    i = check_mensagem(i+1)

#Falha
if random.randrange(0, 2):
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
          telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
          data + "', 'Eca, sai fora')" + ending )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
          telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
          data + "', '0 usuario " + nome + " boqueou Caio Porto')"
          + ending )
else:
    mensagem = rand_mensagem_tinder()
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
          telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +

```

```

        data + "', 'Oi gatao, ta livre pra sair hoje?')" + ending
    )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
        telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
        data + "', '" + mensagem + "')" + ending )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefoneCaio + "', '" +
        telefone + "', 'Caio Porto', '" + nome + "', '" + data +
        "', 'Bora, apareco ai as 20')" + ending )

#Engano
else:
    nome = random.choice(nomes) + " " + random.choice(sobrenomes)
    nome2 = random.choice(nomes)
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
        telefoneCaio + "', '" + nome + "', 'Caio Porto', '" + data +
        "', 'Oi " + nome2 + ", como tao as coisas por ai?')" + ending
    )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefoneCaio + "', '" +
        telefone + "', 'Caio Porto', '" + nome + "', '" + data + "',
        'Telefone errado, nao tem ninguem aqui com esse nome')" +
        ending )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
        telefoneCaio + "', '" + nome + "', 'Caio Porto', '" + data +
        "', 'Opa, desculpa')" + ending )

i+=1
count += 1

```

#### D.4 FASE 4

Listing D.4 – Geração SQL Tabelas

```

import random
import datetime

# Metodos de Geracao

nomes = [
    "Andre", "Alice", "Alfredo", "Arthur", "Armando", "Adriana", "
    Adriano", "Alberto",
    "Beatriz", "Bianca", "Breno", "Bruno", "Bruna", "Bob", "Bernardo", "
    Benjamin", "Bento", "Berenice", "Barney", "Barbara",

```

```

"Caio", "Caroline", "Carolina", "Carlor", "Cauan", "Cleberson", "
  Cicerio", "Cecilia", "Davi", "Camila",
"Eduardo", "Eduarda", "Estevam", "Elena", "Elenilce", "Ester", "
  Everton", "Elen", "Elizandra", "Elizete",
"Fabio", "Fabiana", "Fernando", "Fernanda", "Fatima", "Felipe", "
  Flavia",
"Gabriel", "Gabriela", "Gustavo", "Galvao", "Giovani", "Giovana", "
  Gerson",
"Hugo", "Helena", "Helen",
"Igor", "Iasmim", "Ivan", "Isaac", "Ivone",
"Joao", "Joana", "Jonas", "Julio", "Juliana", "Josue", "Jhonatan", "
  Jardel", "Janaina", "Joyce", "Joseli", "Joselaine", "Jasmine", "
  Juan", "Jade",
"Kerolin", "Kauan",
"Leticia", "Lucas", "Luan", "Lilian", "Laura", "Leonardo",
"Marcos", "Maicon", "Marcelo", "Marcela", "Miriam", "Mauricio", "
  Mario", "Mariana",
"Natan", "Natalia",
"Orlando", "Oscar", "Osmar",
"Pedro", "Pietro", "Paula", "Paulo", "Pamela", "Paloma", "Patricia", "
  Patricio", "Penelope", "Priscila",
"Renato", "Renata", "Rodrigo", "Rubens", "Ricardo", "Roberta",
"Susan", "Savio", "Suelen", "Silmar",
"Tuane", "Thiago", "Tatiana",
"Victor", "Viviane", "Vivian", "Velma", "Valkiria",
"Yasmim"

```

```
]
```

```
sobrenomes = [
```

```

"Sousa", "Souza", "Sant Anna", "Brum", "Goncalves", "Ferreira", "
  Nascimento", "do Nascimento", "Santos", "dos Santos", "de Sant
  Anna", "Ribeiro", "Pereira",
"Outeiro", "Campos", "Matos", "Silva", "da Silva", "Arnaud", "Scholl
  ", "Rodrigues", "Carvalho", "Luna", "Lima", "de Lima", "de
  Carvalho", "Gomes", "Oliveira",
"Alves", "Martins", "Cerqueira", "Barros", "Barbosa", "Andrade", "
  Batista", "Cardoso", "Castro", "Costa", "Dias", "Duarte", "
  Freitas", "Garcia", "Lopes", "Felix",
"Fernandes", "Machado", "Marques", "Medeiros", "Melo", "Monteiro", "
  Moreira", "Nunes", "Oliveira", "Moura", "Reis", "Rocha", "
  Teixeira", "Vieira"

```

```
]
```

```
nomesEmpresas = [
```

```

"Dope Entregas", "Legendary Entregas", "Mercy Entregas", "Burst
  Entregas", "Prospect Entregas", "Entregasex", "Spin Entregas", "
  Box Entregas", "Spirit Entregas", "Central Entregas", "

```

```

    Entregasadora",
    "Metro Entregas", "Creative Entregas", "Boizoid", "Boicog", "Boiish",
    "Boiscape", "Boiporium", "Boinest", "Boiopolis", "Boitastic", "
    Boiiva", "Boiado", "Boiify", "Boique", "Boihut",
    "Boister", "Boizen", "Boinetic", "Boiadil", "Boigenics"
]

matriculas_usadas = {'152354': True, '155623': True, '163224': True}

def nova_matricula():
    value = random.randrange(1, 20000)
    while str(value) in matriculas_usadas:
        value = random.randrange(1, 20000)
    matriculas_usadas[str(value)] = True
    return str(value)

def rand_data_inicial():
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11",
    "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10",
    "11", "12"]
    dia = random.randrange(1, 31)

    if dia == 31:
        mes = random.choice(meses31)
    elif dia > 28:
        mes = random.choice(meses30)
    else:
        mes = random.choice(meses28)
    ano = random.randrange(1994, 2012)
    return str(ano) + "-" + mes + "-" + str(dia), dia, mes, ano

def rand_data_final(diaInicial, mesInicial, anoInicial):
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11",
    "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10",
    "11", "12"]

    while mesInicial > meses31[0]:
        meses31.pop(0)
    while mesInicial > meses30[0]:
        meses30.pop(0)
    while mesInicial > meses28[0]:
        meses28.pop(0)

```

```
dia = random.randrange(diaInicial, 31)

if dia == 31:
    mes = random.choice(meses31)
elif dia > 28:
    mes = random.choice(meses30)
else:
    mes = random.choice(meses28)
ano = random.randrange(anoInicial, 2022)

return str(ano) + "-" + mes + "-" + str(dia)

def rand_data_intervalo(diaInicial, mesInicial, anoInicial, diaFinal,
mesFinal, anoFinal):
    if anoInicial == anoFinal:
        ano = anoInicial
        if mesInicial == mesFinal:
            mes = mesInicial
            dia = random.randrange(diaInicial, diaFinal)
        else:
            mes = random.randrange(mesInicial, mesFinal+1)
            if mes == mesInicial:
                dia = random.randrange(diaInicial, 28)
            elif mes == mesFinal:
                dia = random.randrange(1, diaFinal)
            else:
                dia = random.randrange(1, 28)
    else:
        ano = random.randrange(anoInicial, anoFinal+1)
        if ano == anoInicial:
            mes = random.randrange(mesInicial, 13)
        elif ano == anoFinal:
            mes = random.randrange(1, mesFinal)
        else:
            mes = random.randrange(1, 13)

    if mes == mesInicial:
        dia = random.randrange(diaInicial, 28)
    elif mes == mesFinal:
        dia = random.randrange(1, diaFinal)
    else:
        dia = random.randrange(1, 28)

    if mes < 10:
        mes = "0" + str(mes)
```

```

return str(ano) + "-" + str(mes) + "-" + str(dia)

def rand_motivo_desligamento():
    motivos = ['baixo rendimento', 'perfil nao se enquadra', 'justa
        causa', 'saida voluntaria', 'abandono', 'violacao de conduta', '
        negligencia na funcao']
    return random.choice(motivos)

def rand_nivel():
    nivel = ['leve', 'medio', 'grave']
    return random.choice(nivel)

def rand_incidente_leve():
    incidente = ['Alguma equipe tomou todo o cafe e nao o repos', '
        Alguem sujou completamente o banheiro e desapareceu', 'Haviam
        pessoas conversando extremamente alto no corredor']
    return random.choice(incidente)

def rand_incidente_medio():
    nomeAlvo = random.choice(nomes) + " " + random.choice(sobrenomes)
    nomeAgente = random.choice(nomes) + " " + random.choice(sobrenomes)

    incidente = ['O(A) funcionario(a) ' + nomeAgente + ' estava gritando
        com o(a) fucionario(a) ' + nomeAlvo, 'O(A) funcionario(a) ' +
        nomeAgente + ' estava fazendo bullying com o(a) fucionario(a) ' +
        nomeAlvo]
    return random.choice(incidente)

def rand_incidente_grave():
    nomeAlvo = random.choice(nomes) + " " + random.choice(sobrenomes)

    funcionario = random.choice(funcionarios)
    nomeAgente = funcionario[1] + " " + funcionario[2]

    incidente = ['O(A) funcionario(a) ' + nomeAgente + ' agrediu
        fisicamente o(a) fucionario(a) ' + nomeAlvo, 'O(A) funcionario(a)
        ' + nomeAgente + ' assediou o(a) fucionario(a) ' + nomeAlvo]
    return random.choice(incidente)

def rand_telephone():
    return "(21) 9" + str(random.randrange(6, 9)) + str(random.randrange
        (100, 999)) + "-" + str(random.randrange(1000, 9999))

def rand_mensagem_telemarketing():
    mensagens = ['Ola, voce acaba de ganhar 1000GB de internet, e so
        responder 1 para ativar isso tudo', 'Temos uma promocao
        imperdivel pra voce, ligue para (11)11111111 para saber mais',

```

```

        'Que tal turbinar a sua internet, responda com EU QUERO
        para ativar', 'Somos do do grupo da libertacao liberta,
        e queremos sua ajuda, ligue para (22)22222222 para fazer
        sua doacao']
    return random.choice(mensagens)

def rand_mensagem_tinder():
    mensagens = ['Tem um barzinho aqui pert de casa, quer vir?', 'Que
    tal irmos no shopping amanha?',
    'Vamos num cineminha, adoro o escurinho', 'Fiquei sabendo
    de uma balada nova na cidade, ta afim de ir?']
    return random.choice(mensagens)

# Dados de Desligamento

funcionarios = [
    ('34895', 'Felicia', 'Castro'),
    ('26849', 'Marcos', 'Costa'),
    ('30012', 'Lucas', 'Silveira')
]

print("INSERT INTO Desligamentos(matricula, nome, sobrenome,
    data_entrada, data_saida, motivo_desligamento) \nVALUES")
print("    ('34895', 'Felicia', 'Castro', '2015-06-29', '2022-02-15', '
    nao declarado'),")
print("    ('26849', 'Marcos', 'Costa', '2003-03-17', '2022-02-17', '
    perfil nao se enquadra'),")
print("    ('30012', 'Lucas', 'Silveira', '2019-09-08', '2022-02-16', '
    baixo rendimento'),")
print("    ('24877', 'Fabio', 'Campos', '2020-09-08', '2022-02-10', '
    agressao'),")

count = 0
total = 100

for i in range(0, total):
    motivo = rand_motivo_desligamento()

    matricula = nova_matricula()
    nome = random.choice(nomes)
    sobrenome = random.choice(sobrenomes)

    funcionarios.append((matricula, nome, sobrenome))

    dataInicial, diaInicial, mesInicial, anoInicial = rand_data_inicial
    ()

```

```

dataFinal = rand_data_final(diaInicial, mesInicial, anoInicial)

ending = ','
if count == total - 1:
    ending = ';'

print("      ('"+ matricula + "', '" + nome + "', '" + sobrenome + "',
      '" + dataInicial + "', '" + dataFinal + "', '" + motivo + "')" +
      ending )
count += 1

# Incidentes

print("INSERT INTO Incidentes(id, nome, sobrenome, nivel, reclamacao) \
VALUES")
incidentes = {2: "      ('2', 'Felicia', 'Castro', 'grave', '0 funcionario
      Caio Porto proferiu falas machistas para a funcionaria Felicia
      Castro, falando que a mesma devia ir lavar uma louca em vez de
      reclamar de salario'),"}
incidentes[6] = "      ('6', 'Felicia', 'Castro', 'grave', '0 funcionario
      Caio Porto pediu para ver os seios da funcionaria Felicia Castro'),"
incidentes[8] = "      ('8', 'Fabio', 'Campos', 'leve', 'Roubaram o bife
      da marmitta de Fabio Campos, o mesmo diz achar que foi Davi Duarte'),"
incidentes[12] = "      ('12', 'Davi', 'Duarte', 'leve', 'Davi Duarte
      alega que Fabio Campos instalou uma extensao do Nicholas Cage em seu
      navegador, o mesmo diz ser retaliacao pelo seu bife roubado'),"
incidentes[13] = "      ('13', 'Felicia', 'Castro', 'grave', '0
      funcionario Caio Porto trancou a fucionaria Felicia Castro em uma
      sala dizendo que so a deixaria sair quando ela tirasse a roupa, o
      incidente nao foi mais longe devido a chegada de colegas de Felicia')
      ,"
incidentes[15] = "      ('15', 'Fabio', 'Campos', 'leve', 'Novamente
      roubaram o bife da marmitta de Fabio Campos'),"
incidentes[16] = "      ('16', 'Davi', 'Duarte', 'medio', 'Fabio Campos
      gritou com Davi Duarte por achar que o mesmo roubou seu bife'),"
incidentes[22] = "      ('22', 'Fabio', 'Campos', 'leve', 'Novamente
      roubaram o bife da marmitta de Fabio Campos, o mesmo diz que seu pote
      estava completamente revirado e diz que Davi Duarte estava rindo dele
      '), "
incidentes[25] = "      ('25', 'Davi', 'Duarte', 'grave', 'Davi Duarte
      alega que Fabio Campos o atacou e quebrou dois dentes de Davi apos o
      mesmo ter rido de Fabio ter se irritado com a geladeira por algum
      motivo'),"
incidentes[27] = "      ('27', 'Felicia', 'Castro', 'grave', '0
      funcionario Caio Porto tentou beijar a funcinaria Felicia sem sua
      permissao'),"

```

```

incidentes[28] = "      ('28', NULL, NULL, 'leve', ' O funcionario Lucas
      Silveira passa grande parte do tempo na maquina de cafe sem trabalhar
      '), "

count = 0
total = 40

for i in range(0, total):
    if i in incidentes:
        print(incidentes[i])
        count += 1
        continue

    nivel = rand_nivel()

    if nivel == 'leve':
        reclamacao = rand_incidente_leve()
    elif nivel == 'medio':
        reclamacao = rand_incidente_medio()
    elif nivel == 'grave':
        reclamacao = rand_incidente_grave()

    ending = ','
    if count == total - 1:
        ending = ';'

    if random.randrange(0, 2):
        funcionario = random.choice(funcionarios)
        nome = funcionario[1]
        sobrenome = funcionario[2]
        print("      ('"+ str(i) + "', '" + nome + "', '" + sobrenome + "
            ', '" + nivel + "', '" + reclamacao + "')" + ending )
    else:
        print("      ('"+ str(i) + "', NULL, NULL, '" + nivel + "', '" +
            reclamacao + "')" + ending )

    count += 1

# Mensagens

def check_mensagem(i):
    while True:
        if i in mensagens:
            print(mensagens[i])
            i += 1
            continue

```

```

        else:
            break
    return i

telefoneCaio = "(21)987654321"
dia2015 = "'2015-07-02'"
dia13 = "'2022-05-13'"
dia15 = "'2022-05-15'"
dia16 = "'2022-05-16'"
dia18 = "'2022-05-18'"
dia20 = "'2022-05-20'"

print("INSERT INTO Mensagens(id, telefone_de_origem, telefone_de_destino
    , nome_origem, nome_destino, data, mensagem) \nVALUES")
mensagens = {23: " ('23', '(21)123456789', '(21)987654321', 'Felicia
    Castro', 'Caio Porto', " + dia2015 + " , 'Oi Caio, tudo bem?')",}
mensagens[24] = " ('24', '(21)987654321', '(21)123456789', 'Caio
    Porto', 'Felicia Castro', " + dia2015 + " , 'Oi quem e voce?')",
mensagens[25] = " ('25', '(21)123456789', '(21)987654321', 'Felicia
    Castro', 'Caio Porto', " + dia2015 + " , 'Sou a Felicia, la da
    empresa, entrei esse dias, espero que a gente se de bem')",
mensagens[26] = " ('26', '(21)987654321', '(21)123456789', 'Caio
    Porto', 'Felicia Castro', " + dia2015 + " , 'Ah prazer Felicia,
    espero que a gente se de muito bem mesmo :)')",

mensagens[105] = " ('105', '(21)987654321', '(21)123456789', 'Caio
    Porto', 'Felicia Castro', '2020-02-10' , 'Oi Felicia, tava pensando
    aqui, que tal sairmos pra beber alguma coisa amanha?')",
mensagens[106] = " ('106', '(21)123456789', '(21)987654321', 'Felicia
    Castro', 'Caio Porto', '2020-02-10' , 'Caio, ja falei pra voce, nao
    quero sair com vc, para de me encher o saco')",

mensagens[180] = " ('180', '(21)951753486', '(21)987654321', 'Lucas
    Silveira', 'Caio Porto', " + dia13 + " , 'Boa tarde senhor Caio, tudo
    bem?')",
mensagens[181] = " ('181', '(21)987654321', '(21)951753486', 'Caio
    Porto', 'Lucas Silveira', " + dia13 + " , 'Oi Lucas, fala ai')",
mensagens[182] = " ('182', '(21)951753486', '(21)987654321', 'Lucas
    Silveira', 'Caio Porto', " + dia13 + " , 'Senhor Caio, gostaria de
    tirar uma duvida como o senhor sobre o relatorio de ontem')",
mensagens[183] = " ('183', '(21)951753486', '(21)987654321', 'Lucas
    Silveira', 'Caio Porto', " + dia13 + " , 'O senhor poderia vir na
    minha mesa?')",
mensagens[184] = " ('184', '(21)987654321', '(21)951753486', 'Caio
    Porto', 'Lucas Silveira', " + dia13 + " , 'Ja sai hoje Lucas, amanha
    nos vemos')",

```

```
mensagens [205] = " ('205', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Boa noite senhor Caio, tudo
bem?'),"
mensagens [206] = " ('206', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Acho que senhor se esqueceu
de me ajudar com aquele relatorio'),"
mensagens [207] = " ('207', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Preciso entrega-lo hoje
ainda, voce poderia me ajudar aqui?'),"
mensagens [208] = " ('208', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia15 + " , 'Lucas, voce devia ter me
lembrado antes, eu ja sai do escritorio'),"
mensagens [209] = " ('209', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia15 + " , 'Faca o que voce achar
certo'),"

mensagens [210] = " ('210', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia16 + " , 'Oi Felicia, fiquei sabendo
do que aconteceu com vc, que pena nao e mesmo'),"
mensagens [211] = " ('211', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia16 + " , 'Para de falsidade Caio, eu
sei que foi voce que pediu minha demissao'),"
mensagens [212] = " ('212', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia16 + " , 'Nao linda, que isso,
jamais perderia a chance de poder ver vc todos os dias'),"
mensagens [213] = " ('213', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia16 + " , 'Ah, vai se fuder Caio'),"

mensagens [214] = " ('214', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Oi Caio'),"
mensagens [215] = " ('215', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Tudo bem?'),"
mensagens [216] = " ('216', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Fala Marcos, tudo tranquilo
e ai?'),"
mensagens [217] = " ('217', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'To bem cara, queria saber se a
Vivian ainda vai demorar'),"
mensagens [218] = " ('218', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Sabe me dizer?'),"
mensagens [219] = " ('219', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Perai, vou procurar ela pra
saber'),"
mensagens [220] = " ('220', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Cara, ela disse que deve
estar saindo em mais ou menos 30 minutos'),"
```

```
mensagens [221] = " ('221', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Blz, brigadao cara'),"
mensagens [222] = " ('222', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Nada imagina'),"

mensagens [250] = " ('250', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Caio, bem que todo mundo
fala que voce e um babaca mesmo'),"
mensagens [251] = " ('251', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Te pedi ajuda, vc se
esqueceu de mim, e ainda me demite por incompetencia'),"
mensagens [252] = " ('252', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Vai se fuder ein'),"
mensagens [253] = " ('253', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia16 + " , 'Ue e e culpa minha voce
nao saber fazer o seu trabalho?'),"
mensagens [254] = " ('254', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Vai tomar no cu'),"
mensagens [255] = " ('255', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'O usuario Lucas Silverira
bloqueou Caio porto'),"
mensagens [256] = " ('256', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia16 + " , 'Otario'),"

mensagens [262] = " ('262', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'KRL CAIO, e SeRIO ISSO?! A
VIVAN e MINHA ESPOSA CARA'),"
mensagens [263] = " ('263', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'Como voce pode fazer isso cara
?!'),"
mensagens [264] = " ('264', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'Krl, vc ta fudido'),"
mensagens [265] = " ('265', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'ta muito fudido'),"
mensagens [266] = " ('266', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia18 + " , 'O usuario Caio Porto
bloqueou Marcos Costa'),"

mensagens [267] = " ('267', '(21)123456789', '(21)964875938', 'Felicia
Castro', 'Launa Pascoa', " + dia18 + " , 'Lu, acabei de chegar em
casa, nao to achando meu celular por nada, e nem to conseguindo
deslogar do app'),"
mensagens [268] = " ('268', '(21)964875938', '(21)123456789', 'Launa
Pascoa', 'Felicia Castro', " + dia18 + " , 'Caramba amiga, e agora
??'),"
mensagens [269] = " ('269', '(21)123456789', '(21)964875938', 'Felicia
Castro', 'Launa Pascoa', " + dia18 + " , 'Agora e torcer pra quem
```

```
tiver com o celular nao fazer nada')),"
```

```
mensagens[270] = " ('270', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Sabe Caio, andei pensando
aqui')),"
```

```
mensagens[271] = " ('271', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Agora que to sem nada pra
fazer')),"
```

```
mensagens[272] = " ('272', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'O que vc acha de sairmos pra
ir no shopping na sexta a noite?')),"
```

```
mensagens[273] = " ('273', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia18 + " , 'O que??!! e serio isso?')),
"
```

```
mensagens[274] = " ('274', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Claro que e serio ue, nao
tenho nada pra fazer mesmo, talvez possamos nos conhecer melhor')),"
```

```
mensagens[275] = " ('275', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia18 + " , 'Ta bom entao, sexta quando
eu sair do trampo pode ser? umas 20h?')),"
```

```
mensagens[276] = " ('276', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Marcado entao')),"
```

```
mensagens[284] = " ('284', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia20 + " , 'Oi Caio ja to aqui te
esperando com uma surpresinha :p')),"
```

```
mensagens[285] = " ('285', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro'," + dia20 + " , 'Oi Linda, ja to chegando,
so mais uns 10 minutinhos')),"
```

```
mensagens[291] = " ('291', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia20 + " , 'Cheguei, onde vce ta?')),"
```

```
mensagens[292] = " ('292', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia20 + " , 'To no banheiro do bloco 13')
,)"
```

```
mensagens[293] = " ('293', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia20 + " , 'Bloco 13?')),"
```

```
mensagens[294] = " ('294', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia20 + " , 'Mas o bloco 13 nao ta
fechado pra obra?')),"
```

```
mensagens[295] = " ('295', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia20 + " , 'Eu disse que tinha uma
surpresinha pra vc nao disse?')),"
```

```
mensagens[296] = " ('296', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia20 + " , 'Acho que voce vai querer
essa surpresa :3')),"
```

```

mensagens[297] = "      ('297', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro'," + dia20 + " , 'Epaaa, hoje tem!!'),"

mensagens[298] = "      ('298', '(21)259748631', '(21)987654321', 'Mamae',
      'Caio Porto', " + dia20 + " , 'Cacaoio, pq vc nao esta me atendendo
      ?'),"
mensagens[299] = "      ('299', '(21)259748631', '(21)987654321', 'Mamae',
      'Caio Porto', '2022-05-21' , 'Cacaoio? to preocupada'),"

count = 0
total = 299
i = 0
while i < total:
    i = check_mensagem(i)

    if i < 23:
        data = rand_data_intervalo(1,1,2015,2,7,2015)
    elif i < 105:
        data = rand_data_intervalo(2,7,2015,10,2,2020)
    elif i < 180:
        data = rand_data_intervalo(10,2,2020,13,5,2022)
    elif i < 204:
        data = rand_data_intervalo(13,5,2022,15,5,2022)
    elif i < 250:
        data = rand_data_intervalo(15,5,2022,16,5,2022)
    elif i < 262:
        data = rand_data_intervalo(16,5,2022,18,5,2022)
    elif i < 284:
        data = rand_data_intervalo(18,5,2022,20,5,2022)
    elif i < 291:
        data = "2022-05-20"

    telefone = rand_telephone()

    ending = ','
    if count == total - 1:
        ending = ';'

    # Telemarketing
    if random.randrange(0, 3) == 0:
        mensagem = rand_mensagem_telemarketing()
        print("      ('"+ str(i) + "', '" + telefone + "', '" +
            telefoneCaio + "', 'Telemarketing', 'Caio Porto', '" + data +
            "', '" + mensagem + "')" + ending )
        i = check_mensagem(i+1)

```

```

print("      ('"+ str(i) + "', '" + telefoneCaio + "', '" +
      telefone + "', 'Caio Porto', 'Telemarketing', '" + data + "',
      '0 usuario Caio Porto bloqueou Telemarketing')" + ending )

#Tinder
elif random.randrange(0, 2):
    nome = random.choice(nomes) + " " + random.choice(sobrenomes)
    print("      ('"+ str(i) + "', '" + telefoneCaio + "', '" +
      telefone + "', 'Caio Porto', '" + nome + "', '" + data + "',
      'Oi, tudo bem com vc pessoa linda')" + ending )
    i = check_mensagem(i+1)

#Falha
if random.randrange(0, 2):
    print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
      data + "', 'Eca, sai fora')" + ending )
    i = check_mensagem(i+1)
    print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
      data + "', '0 usuario " + nome + " boqueou Caio Porto')"
      + ending )
else:
    mensagem = rand_mensagem_tinder()
    print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
      data + "', 'Oi gatao, ta livre pra sair hoje?')" + ending
      )
    i = check_mensagem(i+1)
    print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
      data + "', '" + mensagem + "')" + ending )
    i = check_mensagem(i+1)
    print("      ('"+ str(i) + "', '" + telefoneCaio + "', '" +
      telefone + "', 'Caio Porto', '" + nome + "', '" + data +
      "', 'Bora, apareco ai as 20')" + ending )

#Engano
else:
    nome = random.choice(nomes) + " " + random.choice(sobrenomes)
    nome2 = random.choice(nomes)
    print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" + data +
      "', 'Oi " + nome2 + ", como tao as coisas por ai?')" + ending
      )
    i = check_mensagem(i+1)
    print("      ('"+ str(i) + "', '" + telefoneCaio + "', '" +
      telefone + "', 'Caio Porto', '" + nome + "', '" + data + "',

```

```

        'Telefone errado, nao tem ninguem aqui com esse nome')" +
        ending )
    i = check_mensagem(i+1)
    print("      ('"+ str(i) + "', '" + telefone + "', '" +
        telefoneCaio + "', '" + nome + "', 'Caio Porto', '" + data +
        "', 'Opa, desculpa')" + ending )
i+=1
count += 1

```

## D.5 FASE 5

### Listing D.5 – Geração SQL Tabelas

```

import random
import datetime

# Metodos de Geracao

nomes = [
    "Andre", "Alice", "Alfredo", "Arthur", "Armando", "Adriana", "
    Adriano", "Alberto",
    "Beatriz", "Bianca", "Breno", "Bruno", "Bruna", "Bob", "Bernardo", "
    Benjamin", "Bento", "Berenice", "Barney", "Barbara",
    "Caio", "Caroline", "Carolina", "Carlora", "Cauan", "Cleberson", "
    Cicero", "Cecilia", "Davi", "Camila",
    "Eduardo", "Eduarda", "Estevam", "Elena", "Elenilce", "Ester", "
    Everton", "Elen", "Elizandra", "Elizete",
    "Fabio", "Fabiana", "Fernando", "Fernanda", "Fatima", "Felipe", "
    Flavia",
    "Gabriel", "Gabriela", "Gustavo", "Galvao", "Giovani", "Giovana", "
    Gerson",
    "Hugo", "Helena", "Helen",
    "Igor", "Iasmim", "Ivan", "Isaac", "Ivone",
    "Joao", "Joana", "Jonas", "Julio", "Juliana", "Josue", "Jhonatan", "
    Jardel", "Janaina", "Joyce", "Joseli", "Joselaine", "Jasmine", "
    Juan", "Jade",
    "Kerolin", "Kauan",
    "Leticia", "Lucas", "Luan", "Lilian", "Laura", "Leonardo",
    "Marcos", "Maicon", "Marcelo", "Marcela", "Miriam", "Mauricio", "
    Mario", "Mariana",
    "Natan", "Natalia",
    "Orlando", "Oscar", "Osmar",
    "Pedro", "Pietro", "Paula", "Paulo", "Pamela", "Paloma", "Patricia", "
    Patricio", "Penelope", "Priscila",
    "Renato", "Renata", "Rodrigo", "Rubens", "Ricardo", "Roberta",
    "Susan", "Savio", "Suelen", "Silmar",

```

```

    "Tuane", "Thiago", "Tatiana",
    "Victor", "Viviane", "Vivian", "Velma", "Valkiria",
    "Yasmim"
]

sobrenomes = [
    "Sousa", "Souza", "Sant Anna", "Brum", "Goncalves", "Ferreira", "
    Nascimento", "do Nascimento", "Santos", "dos Santos", "de Sant
    Anna", "Ribeiro", "Pereira",
    "Outeiro", "Campos", "Matos", "Silva", "da Silva", "Arnaud", "Scholl
    ", "Rodrigues", "Carvalho", "Luna", "Lima", "de Lima", "de
    Carvalho", "Gomes", "Oliveira",
    "Alves", "Martins", "Cerqueira", "Barros", "Barbosa", "Andrade", "
    Batista", "Cardoso", "Castro", "Costa", "Dias", "Duarte", "
    Freitas", "Garcia", "Lopes", "Felix",
    "Fernandes", "Machado", "Marques", "Medeiros", "Melo", "Monteiro", "
    Moreira", "Nunes", "Oliveira", "Moura", "Reis", "Rocha", "
    Teixeira", "Vieira"
]

nomesEmpresas = [
    "Dope Entregas", "Legendary Entregas", "Mercy Entregas", "Burst
    Entregas", "Prospect Entregas", "Entregasex", "Spin Entregas", "
    Box Entregas", "Spirit Entregas", "Central Entregas", "
    Entregasadora",
    "Metro Entregas", "Creative Entregas", "Boizoid", "Boicog", "Boiish"
    , "Boiscape", "Boiporium", "Boinest", "Boiopolis", "Boitastic", "
    Boiiva", "Boiado", "Boiify", "Boique", "Boihut",
    "Boister", "Boizen", "Boinetic", "Boiadil", "Boigenics"
]

matriculas_usadas = {'152354': True, '155623': True, '163224': True}

def nova_matricula():
    value = random.randrange(1, 20000)
    while str(value) in matriculas_usadas:
        value = random.randrange(1, 20000)
    matriculas_usadas[str(value)] = True
    return str(value)

def rand_data_inicial():
    meses31 = ["01", "03", "05", "07", "08", "10", "12"]
    meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11
    ", "12"]
    meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10
    ", "11", "12"]
    dia = random.randrange(1, 31)

```

```

if dia == 31:
    mes = random.choice(meses31)
elif dia > 28:
    mes = random.choice(meses30)
else:
    mes = random.choice(meses28)
ano = random.randrange(1994, 2012)
return str(ano) + "-" + mes + "-" + str(dia), dia, mes, ano

def rand_data_final(diaInicial, mesInicial, anoInicial):
meses31 = ["01", "03", "05", "07", "08", "10", "12"]
meses30 = ["01", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12"]
meses28 = ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12"]

while mesInicial > meses31[0]:
    meses31.pop(0)
while mesInicial > meses30[0]:
    meses30.pop(0)
while mesInicial > meses28[0]:
    meses28.pop(0)

dia = random.randrange(diaInicial, 31)

if dia == 31:
    mes = random.choice(meses31)
elif dia > 28:
    mes = random.choice(meses30)
else:
    mes = random.choice(meses28)
ano = random.randrange(anoInicial, 2022)

return str(ano) + "-" + mes + "-" + str(dia)

def rand_data_intervalo(diaInicial, mesInicial, anoInicial, diaFinal, mesFinal, anoFinal):
if anoInicial == anoFinal:
    ano = anoInicial
    if mesInicial == mesFinal:
        mes = mesInicial
        dia = random.randrange(diaInicial, diaFinal)
    else:
        mes = random.randrange(mesInicial, mesFinal+1)
        if mes == mesInicial:
            dia = random.randrange(diaInicial, 28)

```

```
        elif mes == mesFinal:
            dia = random.randrange(1, diaFinal)
        else:
            dia = random.randrange(1, 28)

    else:
        ano = random.randrange(anoInicial, anoFinal+1)
        if ano == anoInicial:
            mes = random.randrange(mesInicial, 13)
        elif ano == anoFinal:
            mes = random.randrange(1, mesFinal)
        else:
            mes = random.randrange(1, 13)

    if mes == mesInicial:
        dia = random.randrange(diaInicial, 28)
    elif mes == mesFinal:
        dia = random.randrange(1, diaFinal)
    else:
        dia = random.randrange(1, 28)

    if mes < 10:
        mes = "0" + str(mes)

    return str(ano) + "-" + str(mes) + "-" + str(dia)

def rand_motivo_desligamento():
    motivos = ['baixo rendimento', 'perfil nao se enquadra', 'justa
        causa', 'saida voluntaria', 'abandono', 'violacao de conduta', '
        negligencia na funcao']
    return random.choice(motivos)

def rand_nivel():
    nivel = ['leve', 'medio', 'grave']
    return random.choice(nivel)

def rand_incidente_leve():
    incidente = ['Alguma equipe tomou todo o cafe e nao o repos', '
        Alguem sujou completamente o banheiro e desapareceu', 'Haviam
        pessoas conversando extremamente alto no corredor']
    return random.choice(incidente)

def rand_incidente_medio():
    nomeAlvo = random.choice(nomes) + " " + random.choice(sobrenomes)
    nomeAgente = random.choice(nomes) + " " + random.choice(sobrenomes)
```

```

    incidente = ['O(A) funcionario(a) ' + nomeAgente + ' estava gritando
                com o(a) funcionario(a) ' + nomeAlvo, 'O(A) funcionario(a) ' +
                nomeAgente + ' estava fazendo bullying com o(a) funcionario(a) ' +
                nomeAlvo]
    return random.choice(incidente)

def rand_incidente_grave():
    nomeAlvo = random.choice(nomes) + " " + random.choice(sobrenomes)

    funcionario = random.choice(funcionarios)
    nomeAgente = funcionario[1] + " " + funcionario[2]

    incidente = ['O(A) funcionario(a) ' + nomeAgente + ' agrediu
                fisicamente o(a) funcionario(a) ' + nomeAlvo, 'O(A) funcionario(a)
                ' + nomeAgente + ' assediou o(a) funcionario(a) ' + nomeAlvo]
    return random.choice(incidente)

def rand_telephone():
    return "(21) 9" + str(random.randrange(6, 9)) + str(random.randrange
                (100, 999)) + "-" + str(random.randrange(1000, 9999))

def rand_mensagem_telemarketing():
    mensagens = ['Ola, voce acaba de ganhar 1000GB de internet, e so
                responder 1 para ativar isso tudo', 'Temos uma promocao
                imperdivel pra voce, ligue para (11)11111111 para saber mais',
                'Que tal turbinar a sua internet, responda com EU QUERO
                para ativar', 'Somos do do grupo da libertacao liberta,
                e queremos sua ajuda, ligue para (22)22222222 para fazer
                sua doacao']
    return random.choice(mensagens)

def rand_mensagem_tinder():
    mensagens = ['Tem um barzinho aqui pert de casa, quer vir?', 'Que
                tal irmos no shopping amanha?',
                'Vamos num cineminha, adoro o escurinho', 'Fiquei sabendo
                de uma balada nova na cidade, ta afim de ir?']
    return random.choice(mensagens)

# Dados de Desligamento

funcionarios = [
    ('34895', 'Felicia', 'Castro'),
    ('26849', 'Marcos', 'Costa'),
    ('30012', 'Lucas', 'Silveira')
]

```

```

print("INSERT INTO Desligamentos(matricula, nome, sobrenome,
    data_entrada, data_saida, motivo_desligamento) \nVALUES")
print("    ('34895', 'Felicia', 'Castro', '2015-06-29', '2022-02-15', '
    nao declarado'),")
print("    ('26849', 'Marcos', 'Costa', '2003-03-17', '2022-02-17', '
    perfil nao se enquadra'),")
print("    ('30012', 'Lucas', 'Silveira', '2019-09-08', '2022-02-16', '
    baixo rendimento'),")
print("    ('24877', 'Fabio', 'Campos', '2020-09-08', '2022-02-10', '
    agressao'),")

count = 0
total = 100

for i in range(0, total):
    motivo = rand_motivo_desligamento()

    matricula = nova_matricula()
    nome = random.choice(nomes)
    sobrenome = random.choice(sobrenomes)

    funcionarios.append((matricula, nome, sobrenome))

    dataInicial, diaInicial, mesInicial, anoInicial = rand_data_inicial
        ()
    dataFinal = rand_data_final(diaInicial, mesInicial, anoInicial)

    ending = ','
    if count == total - 1:
        ending = ';'

    print("    ('"+ matricula + "', '" + nome + "', '" + sobrenome + "',
        '" + dataInicial + "', '" + dataFinal + "', '" + motivo + "')" +
        ending )
    count += 1

# Incidentes

print("INSERT INTO Incidentes(id, nome, sobrenome, nivel, reclamacao) \
    nVALUES")
incidentes = {2: "    ('2', 'Felicia', 'Castro', 'grave', '0 funcionario
    Caio Porto proferiu falas machistas para a funcionaria Felicia
    Castro, falando que a mesma devia ir lavar uma louca em vez de
    reclamar de salario'),"}
incidentes[6] = "    ('6', 'Felicia', 'Castro', 'grave', '0 funcionario
    Caio Porto pediu para ver os seios da funcionaria Felicia Castro'),"

```

```

incidentes[8] = "      ('8', 'Fabio', 'Campos', 'leve', 'Roubaram o bife
da marmita de Fabio Campos, o mesmo diz achar que foi Davi Duarte'),"
incidentes[12] = "      ('12', 'Davi', 'Duarte', 'leve', 'Davi Duarte
alega que Fabio Campos instalou uma extensao do Nicholas Cage em seu
navegador, o mesmo diz ser retaliacao pelo seu bife roubado'),"
incidentes[13] = "      ('13', 'Felicia', 'Castro', 'grave', 'O
funcionario Caio Porto trancou a fucionaria Felicia Castro em uma
sala dizendo que so a deixaria sair quando ela tirasse a roupa, o
incidente nao foi mais longe devido a chegada de colegas de Felicia')
,"
incidentes[15] = "      ('15', 'Fabio', 'Campos', 'leve', 'Novamente
roubaram o bife da marmita de Fabio Campos'),"
incidentes[16] = "      ('16', 'Davi', 'Duarte', 'medio', 'Fabio Campos
gritou com Davi Duarte por achar que o mesmo roubou seu bife'),"
incidentes[22] = "      ('22', 'Fabio', 'Campos', 'leve', 'Novamente
roubaram o bife da marmita de Fabio Campos, o mesmo diz que seu pote
estava completamente revirado e diz que Davi Duarte estava rindo dele
'),"
incidentes[25] = "      ('25', 'Davi', 'Duarte', 'grave', 'Davi Duarte
alega que Fabio Campos o atacou e quebrou dois dentes de Davi apos o
mesmo ter rido de Fabio ter se irritado com a geladeira por algum
motivo'),"
incidentes[27] = "      ('27', 'Felicia', 'Castro', 'grave', 'O
funcionario Caio Porto tentou beijar a funcinaria Felicia sem sua
permissao'),"
incidentes[28] = "      ('28', NULL, NULL, 'leve', ' O funcionario Lucas
Silveira passa grande parte do tempo na maquina de cafe sem trabalhar
'),"

count = 0
total = 40

for i in range(0, total):
    if i in incidentes:
        print(incidentes[i])
        count += 1
        continue

    nivel = rand_nivel()

    if nivel == 'leve':
        reclamacao = rand_incidente_leve()
    elif nivel == 'medio':
        reclamacao = rand_incidente_medio()
    elif nivel == 'grave':
        reclamacao = rand_incidente_grave()

```

```

ending = ','
if count == total - 1:
    ending = ';'

if random.randrange(0, 2):
    funcionario = random.choice(funcionarios)
    nome = funcionario[1]
    sobrenome = funcionario[2]
    print("      ('"+ str(i) + "', '" + nome + "', '" + sobrenome + "
        ', '" + nivel + "', '" + reclamacao + "')" + ending )
else:
    print("      ('"+ str(i) + "', NULL, NULL, '" + nivel + "', '" +
        reclamacao + "')" + ending )

count += 1

# Mensagens

def check_mensagem(i):
    while True:
        if i in mensagens:
            print(mensagens[i])
            i += 1
            continue
        else:
            break
    return i

telefoneCaio = "(21)987654321"
dia2015 = "'2015-07-02'"
dia13 = "'2022-05-13'"
dia15 = "'2022-05-15'"
dia16 = "'2022-05-16'"
dia18 = "'2022-05-18'"
dia20 = "'2022-05-20'"

print("INSERT INTO Mensagens(id, telefone_de_origem, telefone_de_destino
    , nome_origem, nome_destino, data, mensagem) \nVALUES")
mensagens = {23: "      ('23', '(21)123456789', '(21)987654321', 'Felicia
    Castro', 'Caio Porto', " + dia2015 + " , 'Oi Caio, tudo bem?')"}
mensagens[24] = "      ('24', '(21)987654321', '(21)123456789', 'Caio
    Porto', 'Felicia Castro', " + dia2015 + " , 'Oi quem e voce?')",
mensagens[25] = "      ('25', '(21)123456789', '(21)987654321', 'Felicia
    Castro', 'Caio Porto', " + dia2015 + " , 'Sou a Felicia, la da
    empresa, entrei esse dias, espero que a gente se de bem'),"

```

```
mensagens [26] = " ('26', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia2015 + " , 'Ah prazer Felicia,
espero que a gente se de muito bem mesmo :)')',"

mensagens [105] = " ('105', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', '2020-02-10' , 'Oi Felicia, tava pensando
aqui, que tal sairmos pra beber alguma coisa amanha?')',"
mensagens [106] = " ('106', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', '2020-02-10' , 'Caio, ja falei pra voce, nao
quero sair com vc, para de me encher o saco')',"

mensagens [180] = " ('180', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia13 + " , 'Boa tarde senhor Caio, tudo
bem?')',"
mensagens [181] = " ('181', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia13 + " , 'Oi Lucas, fala ai')',"
mensagens [182] = " ('182', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia13 + " , 'Senhor Caio, gostaria de
tirar uma duvida como o senhor sobre o relatorio de ontem')',"
mensagens [183] = " ('183', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia13 + " , 'O senhor poderia vir na
minha mesa?')',"
mensagens [184] = " ('184', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia13 + " , 'Ja sai hoje Lucas, amanha
nos vemos')',"

mensagens [205] = " ('205', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Boa noite senhor Caio, tudo
bem?')',"
mensagens [206] = " ('206', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Acho que senhor se esqueceu
de me ajudar com aquele relatorio')',"
mensagens [207] = " ('207', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia15 + " , 'Preciso entrega-lo hoje
ainda, voce poderia me ajudar aqui?')',"
mensagens [208] = " ('208', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia15 + " , 'Lucas, voce devia ter me
lembrado antes, eu ja sai do escritorio')',"
mensagens [209] = " ('209', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia15 + " , 'Faca o que voce achar
certo')',"

mensagens [210] = " ('210', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia16 + " , 'Oi Felicia, fiquei sabendo
do que aconteceu com vc, que pena nao e mesmo')',"
mensagens [211] = " ('211', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia16 + " , 'Para de falsidade Caio, eu
```

```
sei que foi voce que pediu minha demissao'),'',"
mensagens [212] = "      ('212', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia16 + " , 'Nao linda, que isso,
jamais perderia a chance de poder ver vc todos os dias'),'',"
mensagens [213] = "      ('213', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia16 + " , 'Ah, vai se fuder Caio'),'',"

mensagens [214] = "      ('214', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Oi Caio'),'',"
mensagens [215] = "      ('215', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Tudo bem?'),'',"
mensagens [216] = "      ('216', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Fala Marcos, tudo tranquilo
e ai?'),'',"
mensagens [217] = "      ('217', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'To bem cara, queria saber se a
Vivian ainda vai demorar'),'',"
mensagens [218] = "      ('218', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Sabe me dizer?'),'',"
mensagens [219] = "      ('219', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Perai, vou procurar ela pra
saber'),'',"
mensagens [220] = "      ('220', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Cara, ela disse que deve
estar saindo em mais ou menos 30 minutos'),'',"
mensagens [221] = "      ('221', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia16 + " , 'Blz, brigadao cara'),'',"
mensagens [222] = "      ('222', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia16 + " , 'Nada imagina'),'',"

mensagens [250] = "      ('250', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Caio, bem que todo mundo
fala que voce e um babaca mesmo'),'',"
mensagens [251] = "      ('251', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Te pedi ajuda, vc se
esqueceu de mim, e ainda me demite por incompetencia'),'',"
mensagens [252] = "      ('252', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Vai se fuder ein'),'',"
mensagens [253] = "      ('253', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia16 + " , 'Ue e e culpa minha voce
nao saber fazer o seu trabalho?'),'',"
mensagens [254] = "      ('254', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'Vai tomar no cu'),'',"
mensagens [255] = "      ('255', '(21)951753486', '(21)987654321', 'Lucas
Silveira', 'Caio Porto', " + dia16 + " , 'O usuario Lucas Silverira
bloqueou Caio porto'),'',"
```

```
mensagens [256] = " ('256', '(21)987654321', '(21)951753486', 'Caio
Porto', 'Lucas Silveira', " + dia16 + " , 'Otario'),"

mensagens [262] = " ('262', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'KRL CAIO, e SeRIO ISSO?! A
VIVAN e MINHA ESPOSA CARA'),"

mensagens [263] = " ('263', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'Como voce pode fazer isso cara
?!'),"

mensagens [264] = " ('264', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'Krl, vc ta fudido'),"

mensagens [265] = " ('265', '(21)976431852', '(21)987654321', 'Marcos
Costa', 'Caio Porto', " + dia18 + " , 'ta muito fudido'),"

mensagens [266] = " ('266', '(21)987654321', '(21)976431852', 'Caio
Porto', 'Marcos Costa', " + dia18 + " , 'O usuario Caio Porto
bloqueou Marcos Costa'),"

mensagens [267] = " ('267', '(21)123456789', '(21)964875938', 'Felicia
Castro', 'Launa Pascoa', " + dia18 + " , 'Lu, acabei de chegar em
casa, nao to achando meu celular por nada, e nem to conseguindo
deslogar do app'),"

mensagens [268] = " ('268', '(21)964875938', '(21)123456789', 'Launa
Pascoa', 'Felicia Castro', " + dia18 + " , 'Caramba amiga, e agora
??'),"

mensagens [269] = " ('269', '(21)123456789', '(21)964875938', 'Felicia
Castro', 'Launa Pascoa', " + dia18 + " , 'Agora e torcer pra quem
tiver com o celular nao fazer nada'),"

mensagens [270] = " ('270', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Sabe Caio, andei pensando
aqui'),"

mensagens [271] = " ('271', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Agora que to sem nada pra
fazer'),"

mensagens [272] = " ('272', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'O que vc acha de sairmos pra
ir no shopping na sexta a noite?'),"

mensagens [273] = " ('273', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia18 + " , 'O que??!! e serio isso?'),
"

mensagens [274] = " ('274', '(21)123456789', '(21)987654321', 'Felicia
Castro', 'Caio Porto', " + dia18 + " , 'Claro que e serio ue, nao
tenho nada pra fazer mesmo, talvez possamos nos conhecer melhor'),"

mensagens [275] = " ('275', '(21)987654321', '(21)123456789', 'Caio
Porto', 'Felicia Castro', " + dia18 + " , 'Ta bom entao, sexta quando
eu sair do trampo pode ser? umas 20h?'),"
```

```
mensagens[276] = "      ('276', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia18 + " , 'Mercado entao'),"

mensagens[284] = "      ('284', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'Oi Caio ja to aqui te
      esperando com uma surpresinha :p'),"
mensagens[285] = "      ('285', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro'," + dia20 + " , 'Oi Linda, ja to chegando,
      so mais uns 10 minutinhos'),"

mensagens[291] = "      ('291', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro', " + dia20 + " , 'Cheguei, onde vce ta?'),"
mensagens[292] = "      ('292', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'To no banheiro do bloco 13')
      ,"
mensagens[293] = "      ('293', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro', " + dia20 + " , 'Bloco 13?'),"
mensagens[294] = "      ('294', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro', " + dia20 + " , 'Mas o bloco 13 nao ta
      fechado pra obra?'),"
mensagens[295] = "      ('295', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'Eu disse que tinha uma
      surpresinha pra vc nao disse?'),"
mensagens[296] = "      ('296', '(21)123456789', '(21)987654321', 'Felicia
      Castro', 'Caio Porto', " + dia20 + " , 'Acho que voce vai querer
      essa surpresa :3'),"
mensagens[297] = "      ('297', '(21)987654321', '(21)123456789', 'Caio
      Porto', 'Felicia Castro'," + dia20 + " , 'Epaaa, hoje tem!!'),"

mensagens[298] = "      ('298', '(21)259748631', '(21)987654321', 'Mamae',
      'Caio Porto', " + dia20 + " , 'Cacaio, pq vc nao esta me atendendo
      ?'),"
mensagens[299] = "      ('299', '(21)259748631', '(21)987654321', 'Mamae',
      'Caio Porto', '2022-05-21' , 'Cacaio? to preocupada'),"

count = 0
total = 299
i = 0
while i < total:
    i = check_mensagem(i)

    if i < 23:
        data = rand_data_intervalo(1,1,2015,2,7,2015)
    elif i < 105:
        data = rand_data_intervalo(2,7,2015,10,2,2020)
    elif i < 180:
        data = rand_data_intervalo(10,2,2020,13,5,2022)
```

```

elif i < 204:
    data = rand_data_intervalo(13,5,2022,15,5,2022)
elif i < 250:
    data = rand_data_intervalo(15,5,2022,16,5,2022)
elif i < 262:
    data = rand_data_intervalo(16,5,2022,18,5,2022)
elif i < 284:
    data = rand_data_intervalo(18,5,2022,20,5,2022)
elif i < 291:
    data = "2022-05-20"

telefone = rand_telephone()

ending = ','
if count == total - 1:
    ending = ';'

# Telemarketing
if random.randrange(0, 3) == 0:
    mensagem = rand_mensagem_telemarketing()
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
          telefoneCaio + "', 'Telemarketing', 'Caio Porto', '" + data +
          "', '" + mensagem + "')" + ending )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefoneCaio + "', '" +
          telefone + "', 'Caio Porto', 'Telemarketing', '" + data + "',
          '0 usuario Caio Porto bloqueou Telemarketing')" + ending )

#Tinder
elif random.randrange(0, 2):
    nome = random.choice(nomes) + " " + random.choice(sobrenomes)
    print("    ('"+ str(i) + "', '" + telefoneCaio + "', '" +
          telefone + "', 'Caio Porto', '" + nome + "', '" + data + "',
          'Oi, tudo bem com vc pessoa linda')" + ending )
    i = check_mensagem(i+1)

#Falha
if random.randrange(0, 2):
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
          telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
          data + "', 'Eca, sai fora')" + ending )
    i = check_mensagem(i+1)
    print("    ('"+ str(i) + "', '" + telefone + "', '" +
          telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
          data + "', '0 usuario " + nome + " boqueou Caio Porto')"
          + ending )
else:

```

```

mensagem = rand_mensagem_tinder()
print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
      data + "', 'Oi gatao, ta livre pra sair hoje?')") + ending
    )
i = check_mensagem(i+1)
print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" +
      data + "', '" + mensagem + "')") + ending )
i = check_mensagem(i+1)
print("      ('"+ str(i) + "', '" + telefoneCaio + "', '" +
      telefone + "', 'Caio Porto', '" + nome + "', '" + data +
      "', 'Bora, apareco ai as 20')") + ending )

#Engano
else:
    nome = random.choice(nomes) + " " + random.choice(sobrenomes)
    nome2 = random.choice(nomes)
    print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" + data +
      "', 'Oi " + nome2 + ", como tao as coisas por ai?')") + ending
    )
i = check_mensagem(i+1)
print("      ('"+ str(i) + "', '" + telefoneCaio + "', '" +
      telefone + "', 'Caio Porto', '" + nome + "', '" + data + "',
      'Telefone errado, nao tem ninguem aqui com esse nome')") +
      ending )
i = check_mensagem(i+1)
print("      ('"+ str(i) + "', '" + telefone + "', '" +
      telefoneCaio + "', '" + nome + "', 'Caio Porto', '" + data +
      "', 'Opa, desculpa')") + ending )

i+=1
count += 1

```