



**UNIVERSIDADE FEDERAL DO TOCANTINS
CÂMPUS UNIVERSITÁRIO DE PALMAS
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**PROTÓTIPO DE UMA FERRAMENTA GAMIFICADA PARA A APLICAÇÃO DE
ATIVIDADES PRÁTICAS EM SALA DE AULA EM UMA DISCIPLINA DE
INTRODUÇÃO À PROGRAMAÇÃO**

NATÃ WERECLYS BANDEIRA LIMA

PALMAS (TO)

2022

NATÃ WERECLYS BANDEIRA LIMA

PROTÓTIPO DE UMA FERRAMENTA GAMIFICADA PARA A APLICAÇÃO DE
ATIVIDADES PRÁTICAS EM SALA DE AULA EM UMA DISCIPLINA DE
INTRODUÇÃO À PROGRAMAÇÃO

Trabalho de Conclusão de Curso II apresentado
à Universidade Federal do Tocantins para
obtenção do título de Bacharel em Ciência da
Computação, sob a orientação do(a) Prof.(a)
Dr. Edeilson Milhomem da Silva.

Orientador: Dr. Edeilson Milhomem da Silva

PALMAS (TO)

2022

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da Universidade Federal do Tocantins

L732p Lima, Natã Wereclys Bandeira.
Protótipo de uma Ferramenta Gamificada para a Aplicação de
Atividades Práticas em Sala de Aula em uma Disciplina de Introdução
à Programação. / Natã Wereclys Bandeira Lima. – Palmas, TO, 2022.
62 f.

Monografia Graduação - Universidade Federal do Tocantins –
Câmpus Universitário de Palmas - Curso de Ciências da Computação,
2022.

Orientador: Edeilson Milhomem da Silva

1. Gamificação. 2. Motivação. 3. Programação. 4. Jogos. I. Título

CDD 004

TODOS OS DIREITOS RESERVADOS – A reprodução total ou parcial, de
qualquer forma ou por qualquer meio deste documento é autorizado desde
que citada a fonte. A violação dos direitos do autor (Lei nº 9.610/98) é crime
estabelecido pelo artigo 184 do Código Penal.

**Elaborado pelo sistema de geração automática de ficha catalográfica
da UFT com os dados fornecidos pelo(a) autor(a).**

NATÃ WERECLYS BANDEIRA LIMA

PROTÓTIPO DE UMA FERRAMENTA GAMIFICADA PARA A APLICAÇÃO DE
ATIVIDADES PRÁTICAS EM SALA DE AULA EM UMA DISCIPLINA DE
INTRODUÇÃO À PROGRAMAÇÃO

Trabalho de Conclusão de Curso II apresentado à UFT – Universidade Federal do Tocantins – Câmpus Universitário de Palmas, Curso de Ciência da Computação foi avaliado para a obtenção do título de Bacharel e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Data de aprovação: 7 / 2 / 2022

Banca Examinadora:

Prof. Dr. Edeilson Milhomem da Silva

Prof. Dr. Eduardo Ferreira Ribeiro

Profa. Ma. Juliana Leitão Dutra

*Motivação é a arte de fazer as
pessoas fazerem o que você quer
que elas façam porque elas o
querem fazer.*

Dwight Eisenhower

AGRADECIMENTOS

Agradeço primeiramente a Deus, que me deu forças para segui meu caminho e obter minhas conquistas. Ao meu orientador, Edeilson Milhomem da Silva que me concedeu a oportunidade de desenvolver este trabalho. Agradeço também, por sua dedicação em me orientar, e pela paciência durante o desenvolvimento deste trabalho. A minha mãe Auriceia Bandeira Lima, minha avó Maria de Lys Bandeira e outros familiares, que me proveram suporte para conseguir chegar onde cheguei. Agradeço a meus amigos, Éleonn Milhomem, Arthur Gabriel, João Vitor Jacundá e Matheus Aguiar, que muitas vezes me auxiliaram e me deram força para continuar.

RESUMO

Gamificação é o processo de aplicação de elementos de jogos fora do contexto dos jogos, e pode motivar as pessoas a realizarem tarefas. Deste modo, este trabalho tem como objetivo propor uma ferramenta gamificada focada na prática de Introdução à Programação. Para isso, foi desenvolvido um modelo de estrutura gamificada, visando estruturar como os elementos do jogo atuarão no ambiente de uma disciplina de ensino de programação.

Palavra-chave: Gamificação. Motivação. Educação. Programação. Jogos.

ABSTRACT

Gamification is the process of applying game elements outside the context of games, and it can motivate people to perform tasks. This work aims to propose a gamified tool focused on the practice of Introduction to Programming. For this, a gamified structure model was developed, with the objective of structuring how the game elements will act in the environment of a programming teaching discipline.

Keywords: Gamification. Motivation. Education. Programming. Games.

LISTA DE FIGURAS

Figura 1 – Fogg Behavior Model (Modelo comportamental de Fogg)	16
Figura 2 – Levantamento conceitual sobre Gamificação	17
Figura 3 – Pirâmide representando o tripé de um projeto com uso de gamification.	18
Figura 4 – Grupos de perfis de jogadores	21
Figura 5 – Capturas da ferramenta Duolingo	24
Figura 6 – Capturas da ferramenta Klan Academy	25
Figura 7 – Estrutura do MAAGIP	31
Figura 8 – Módulo de Desafios	32
Figura 9 – Módulo de Processamento	34
Figura 10 – Módulo Gamificado	35
Figura 11 – Parâmetros de engajamento do MAAGIP	38
Figura 12 – Badges de bronze	44
Figura 13 – Badges de prata	45
Figura 14 – Badges de ouro	46
Figura 15 – Arquitetura em três camadas do LPC	48
Figura 16 – Ciclo de autenticação	51
Figura 17 – Algoritmo de gamificação do LPC-GAME	55
Figura 18 – Página de fases do protótipo da ferramenta LPC	57
Figura 19 – Página de feed do protótipo da ferramenta LPC	58
Figura 20 – Página de controle de turmas do protótipo da ferramenta LPC	59

LISTA DE TABELAS

Tabela 1 – Associação entre os perfis de estudantes, tipos de diversão e emoções.	23
Tabela 2 – Associação entre os princípios de game design, objetivos e componente de jogos.	40
Tabela 3 – Descrição do <i>status</i> das fases	41
Tabela 4 – Relação da quantidade de tentativas ao resolver desafios com a porcentagem de pontos obtidos da Base de Pontos	42

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Objetivo geral	13
1.2	Objetivos específicos	14
1.3	Justificativa	14
1.4	Organização do trabalho	14
2	REVISÃO BIBLIOGRÁFICA	15
2.1	Motivação e mudança de comportamento	15
2.2	Gamificação	16
2.2.1	Elementos de jogos	17
2.2.2	Categorias de jogadores	21
2.3	Aplicações da gamificação no contexto educacional	24
3	METODOLOGIA	27
3.1	Objetivos gamificados	27
3.2	Implantação do modelo MAGIP	27
3.3	Prototipar o modelo gamificado	29
3.4	Ferramentas usadas	29
4	MODELO DE APLICAÇÃO DE ATIVIDADES GAMIFICADAS DE INTRODUÇÃO A PROGRAMAÇÃO (MAAGIP)	31
4.1	Módulo de Desafios	32
4.2	Módulo de Processamento	34
4.3	Módulo Gamificado	35
4.4	Elementos da Ferramenta Gamificada	36
4.4.1	Cenário (Ambiente)	36

4.4.2	Jogadores (Participantes)	37
4.4.3	Modelo de engajamento dos jogadores	38
4.4.4	Elementos de Jogos	39
4.4.4.1	Níveis	40
4.4.4.2	Pontos	41
4.4.4.3	Badges	43
4.4.4.4	Ranking (Tabela de classificação)	46
4.4.4.5	Avatar (Perfil do aluno)	47
5	FERRAMENTA LPC (LEARN, PLAY AND CODE)	48
5.1	Microsserviço de autenticação(LPC-AUTH)	50
5.2	Microsserviço de repositório de atividades	52
5.3	Microsserviço de sistema gamificado	55
5.4	Layer de apresentação	56
5.4.1	Interface gamificada da LPC	57
5.4.2	Interface de controle de turmas	58
6	CONCLUSÃO	60
	REFERÊNCIAS	61

1 INTRODUÇÃO

A alta taxa de desistência de alunos em cursos como o de Ciência da Computação demonstra um grande desafio a ser enfrentado pelas instituições de ensino (SCHOEFFEL et al., 2018). Nas áreas de ensino de software, inúmeros motivos podem causar a desistência dos alunos, dentre eles, está a dificuldade em programação de computadores. Introdução à programação (IP) é uma das disciplinas essenciais nas grades dos cursos de computação, já que para realizar atividades mais complexas como modelagem de software, estruturas de dados e algoritmos mais complexos, é demandado um conhecimento sólido em lógica de programação. Entretanto, mesmo com a importância do estudo de IP, a disciplina apresenta evasão ou reprovação por parte dos alunos (ROCHA et al., 2010).

Uma pesquisa realizada na USP mostrou que a disciplina de introdução à programação tem uma taxa de reprovação ou trancamento de 29.31%. Além disso, no período de 2010 a 2014, 25% dos alunos precisaram realizar a disciplina duas vezes ou mais (BOSSE; GEROSA, 2015). Muitas vezes, as causas de desistências ou reprovações podem estar relacionadas à dificuldade de aprendizado causada pela exigência de conhecimento lógico-matemático, o que pode gerar dificuldade de assimilação dos conteúdos e até um desinteresse por parte dos alunos (ROCHA et al., 2010).

Um dos fatores necessários para o aprendizado de programação é a prática, ou seja, a realização das atividades propostas pelos professores em sala de aula, que se mostram essenciais no processo de desenvolvimento do conhecimento de lógica de programação. Em muitos casos, as atividades são disponibilizadas em forma de listas de exercícios e desafios, que podem ser aplicados em sala de aula ou para serem resolvidas em horário extraclasse. Entretanto, muitos alunos perdem a motivação e não demonstram a devida importância na prática de programação.

Gamificação ou ludificação é o processo de aplicar técnicas e elementos de design de jogos em demais contextos que não sejam um jogo (WERBACH; HUNTER, 2012), e pode ser usado para melhorar a motivação de alunos ao realizarem atividades de IP. É importante ressaltar que a gamificação não implica em criar um jogo em um determinado contexto, e sim, aplicar estratégias e métodos comuns nos jogos em demais ambientes, visando que os participantes sintam-se motivados.

1.1 Objetivo geral

Desenvolvimento de um protótipo de uma ferramenta gamificada para a aplicação de Atividades Práticas da disciplina de Introdução à Programação.

1.2 Objetivos específicos

1. Desenvolver a estrutura ludificada da ferramenta, destacando os objetivos gamificados, resultados esperados e elementos de jogos.
2. Definir o modelo taxonômico para organização das fases do ambiente gamificado.
3. Definir as regras do ambiente gamificado para controlar as interações dos usuários com o ambiente: permissões, punições, quantidade de pontos, dificuldade das atividades, nível dos alunos, objetivo do nível, e etc;
4. Desenvolvimento do protótipo da ferramenta gamificada.

1.3 Justificativa

O ambiente do jogo pode ser comparado aos ambientes de aprendizado em muitos aspectos. Ambos criam um espaço onde é fomentado elementos como, competitividade, trabalho em grupo e resolução de desafios. (WANG; LV, 2018) Ainda que a aplicação dos conceitos da gamificação na área da educação seja algo relativamente novo, essa técnica pode gerar muitos resultados positivos, aumentando o engajamento dos alunos. Diversas entidades educacionais estudam formas de aplicar o dinamismo dos jogos em salas de aula, possibilitando que alunos, ao realizarem atividades, apresentem o mesmo nível de interesse que demonstram com jogos (TED, 2010; TOLOMEI, 2017).

Nesta perspectiva, seria interessante a disponibilização de uma ferramenta que possa proporcionar aos alunos que praticam programação, a sensação de diversão que é comumente observada em jogos. Ao usar de elementos de jogos, o processo de aprendizado dos alunos pode se tornar mais atrativo, e talvez como resultado ocorra um melhor entendimento dos conceitos de programação e pensamento lógico-matemático.

1.4 Organização do trabalho

O presente trabalho está dividido em seis capítulos. São eles: (1) contém a introdução do trabalho descrevendo o objetivo geral e os objetivos específicos; (2) traz o referencial teórico com os conceitos adotados neste trabalho; (3) descreve o método usado para desenvolvimento e prototipação da ferramenta gamificada; (4) consiste na construção de um modelo gamificado para aplicação de atividade de introdução a programação, ou seja, o MAAGIP; (5) define a arquitetura e o protótipo da ferramenta LPC, usando como base a estrutura do modelo MAAGIP; (6) e por fim, a conclusão do trabalho.

2 REVISÃO BIBLIOGRÁFICA

2.1 Motivação e mudança de comportamento

Motivação é um processo psicológico que influencia como um indivíduo persiste em uma atividade, que quando bem dirigida pode trazer benefícios à autorrealização (PEDRO, 2016). Manter participantes motivados em uma atividade pode ser complexo, independente da metodologia aplicada. Assim, vários estudos são realizados visando reduzir muitos dos efeitos negativos que interferem no interesse dos participantes ao realizarem tarefas (FREDRICKS; FILSECKER; LAWSON, 2016). Uma pessoa motivada apresenta como comportamento uma disposição natural quando realiza uma tarefa, que gera por si só prazer ao participante (RYAN; DECI, 2002).

A motivação pode ser dividida em duas categorias, sendo a intrínseca e a extrínseca. A motivação extrínseca relaciona-se com a parte externa do indivíduo, movendo o mesmo a realizar uma ação com foco em uma recompensa. O participante pode deixar de realizar alguma atividade, caso não encontre uma possível recompensa, já que valoriza o resultado e não o desenvolvimento, portanto, isso pode indicar que a atividade causa pouca satisfação ao ser realizada.

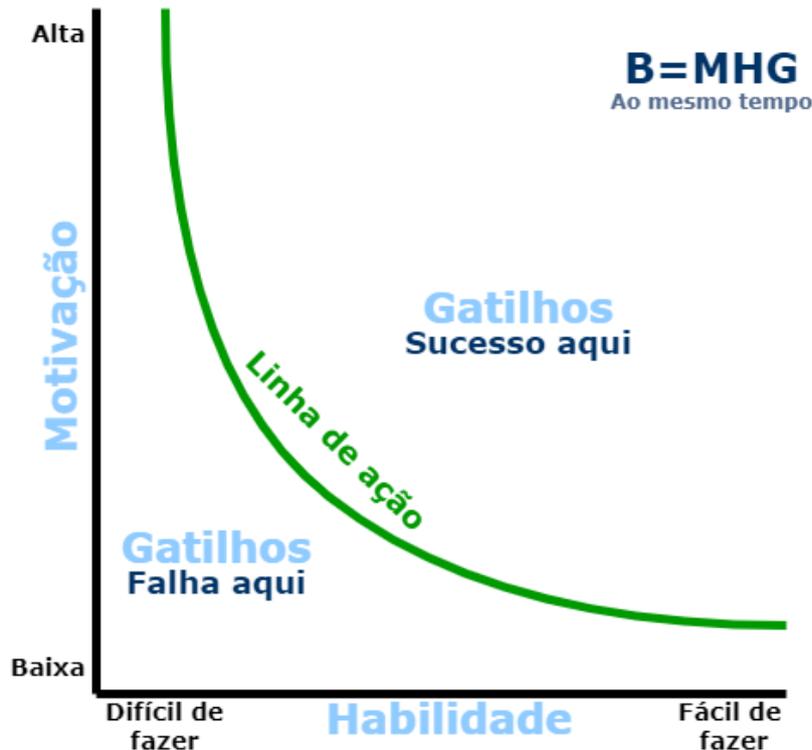
Já a motivação intrínseca origina-se na parte interna do indivíduo, onde os interesses e gostos do indivíduo estão relacionados com a simples ação de realizar uma atividade. Nesse caso, explorar e evoluir as habilidades são grandes oportunidades, independente de haver uma recompensa, já que depende exclusivamente do sujeito (LOMBRISER; VALK, 2014).

Algumas atividades como jogar, passar tempo com os amigos, ler, praticar esportes, são consideradas intrinsecamente motivadas, já que oferecem uma sensação natural de divertimento para muitas pessoas. Nos casos de atividades intrinsecamente motivadas, a aplicação de elementos de recompensa extrínseca pode causar a queda da motivação. Mesmo assim, os elementos de motivação extrínseca têm suas vantagens, e podem ser aplicados sem causar problemas em casos onde a motivação intrínseca é baixa ou até mesmo inexistente (MARINS, 2013).

Mudança de comportamento depende de habilidade e motivação disparadas por um gatilho que impulsiona a alteração de atitude de uma pessoa. Assim, o autor Fogg (2009a) propôs o modelo FBM (*Fogg Behavior Model*) que é exibido na Figura 1, e que define que para ocorrer uma mudança de comportamento, a pessoa deve dispor de três fatores que devem ocorrer de modo simultâneo, sendo estes: (1) motivação, que é um motivo para realizar uma ação; (2) habilidade, que representa a capacidade do indivíduo de realizar uma dada atividade determinada pelo contexto; (3) gatilho, que é qualquer coisa que dispara uma ação comportamental do participante.

Figura 1 – Fogg Behavior Model (Modelo comportamental de Fogg)

Fonte: (FOGG, 2009a)



O FBM (Figura 1) é composto por dois eixos, um dos quais, o vertical, representa a motivação, e o horizontal a habilidade. A linha de ação separa o sucesso da falha que pode ser obtida em alguma atividade. O FBM define que para atingir um comportamento alvo é preciso altos níveis de motivação e habilidade. Assim, um usuário com alta habilidade e baixa motivação precisa de um aumento motivacional para atingir o comportamento desejado (FOGG, 2009b). De modo que, quando um usuário está motivado a realizar uma ação, mesmo que difícil, será capaz de atingir um resultado. Entretanto, até mesmo uma tarefa simples pode se tornar complexa, com um baixo nível de motivação (WEISER et al., 2015/09).

2.2 Gamificação

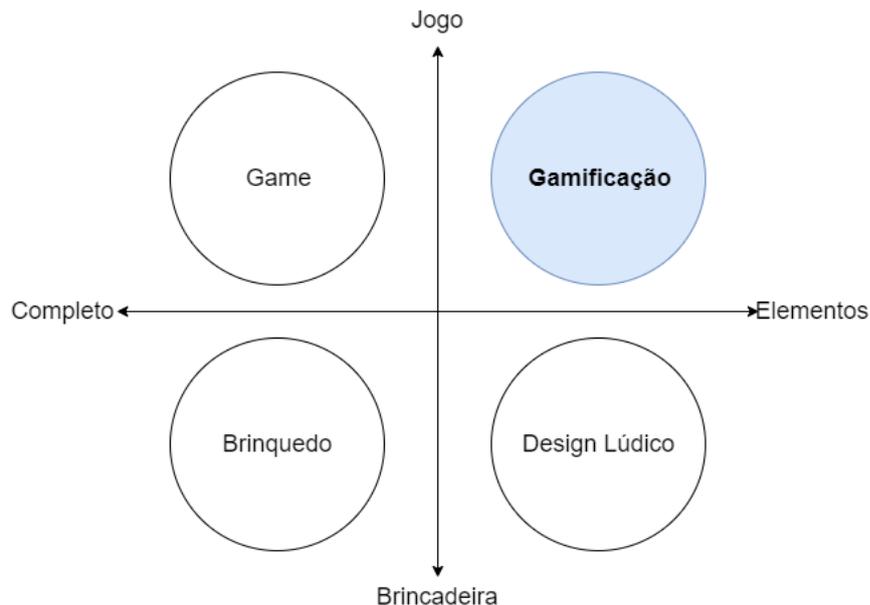
Gamificação ou ludificação é o uso de elementos de jogos em contextos não associados aos mesmos. Em outras palavras, é a utilização de elementos e lógica de jogos em ambientes onde não são comumente aplicados (DETERDING et al., 2011). Enquanto jogos habituais usam game design para fins de entretenimento, a gamificação adota apenas os elementos de jogos. Essa adoção de elementos pode melhorar a experiência do usuário e a motivação ao realizar atividades (CAHYANI, 2016).

Como exibido na Figura 2, tanto jogos como brincadeiras podem ser diferenciados da ludificação através do uso dos componentes que os integram, sejam esses aplicados

em partes ou em uso completo. A definição de gamificação contrasta com o conceito de brincar, se aproximando mais da definição de jogar. Em outras palavras, enquanto um game usa de todos os elementos de um jogo, a gamificação usa algumas partes, ou seja, alguns elementos (DETERDING et al., 2011).

Figura 2 – Levantamento conceitual sobre Gamificação

Fonte: Adaptado de Deterding et al. (2011)



O conceito de usar elementos de jogo para incentivar pessoas a realizarem atividades não havia se tornado convencional até o ano de 2010, em que o termo gamificação ganhou popularidade. Essa popularidade derivou-se devido da crença de que o método possa causar mudanças de comportamento como, maior motivação em tarefas, atitude mais amigável entre pessoas e colaboração entre indivíduos (DICHEV; DICHEVA, 2017).

Apesar de ter se tornado popular por volta de 2010, muitos dos conceitos da gamificação já eram usados antes (PINTO; SILVA, 2017). Atualmente, a técnica é adotada em diversos ambientes diferentes, como marketing, saúde, bem-estar, espaços empresariais, redes sociais e na educação. Devido a isso, um grande número de instituições estudam a aplicação da ludificação, com o intuito de motivar os participantes nas atividades propostas. (GOSHEVSKI; VELJANOSKA; HATZIAPOSTOULOU, 2017).

2.2.1 Elementos de jogos

Elementos de jogos são todas as ferramentas utilizadas na construção de uma aplicação gamificada. Esses elementos são extraídos de jogos e aplicados no contexto que se deseja “ludificar” (ALVES, 2015). A pirâmide exibida na Figura 3 foi desenvolvida por (WERBACH; HUNTER, 2012) e tem como propósito exemplificar a relação dos elemen-

tos gamificados. Tais elementos podem ser agrupados em três categorias relevantes, sendo essas: dinâmica, mecânica e componentes.

Figura 3 – Pirâmide representando o tripé de um projeto com uso de gamification.

Fonte: Adaptado de Werbach e Hunter (2012)



A experiência que se busca alcançar com um sistema gamificado, fica posicionada na extremidade da pirâmide (Figura 3). Cada mecânica está relacionada a uma dinâmica ou mais, e cada componente está relacionado a um conjunto de mecânicas ou mais (WERBACH; HUNTER, 2012). No topo da pirâmide se encontra a dinâmica, que, segundo Alves (2015) é “constituída por elementos responsáveis por atribuir coerência e padrões regulares à experiência”. Além disso, são definidas como o nível mais alto de abstração dos elementos de jogos, ou seja, definem os aspectos gerais do sistema (WERBACH; HUNTER, 2012). A seguir são listadas as principais dinâmicas de jogos:

- **Constrições:** se referem às restrições impostas à liberdade do jogador. Ao longo do caminho, seu esforço em atingir um objetivo enfrenta limitações impostas pelas constrições. Em virtude disso, o participante desenvolve pensamento estratégico ao tentar alcançar seu propósito, já que as constrições o impedem de atingir a solução pelo caminho simples.
- **Emoções:** um jogo pode disparar no jogador uma infinidade de emoções, que podem ir de alegria a tristeza. Entretanto, a gamificação está conectada ao mundo real, ou seja, existe em um contexto diferente de um jogo. Mesmo assim, ainda haverá algumas emoções disparadas pelos feedbacks, ou, quando se atingir algum objetivo.

- Narrativa (*Storytelling*): a narrativa é o que ocasiona a união das demais mecânicas em um contexto único, sendo também responsável pelo sentimento de coerência. Apesar de ser um elemento chave para envolver o jogador, nem todos os jogos apresentam uma narrativa.
- Progressão: mecânica responsável por dar ao jogador a sensação de estar evoluindo no jogo. Isso faz com que o mesmo analise se vale a pena progredir, o que o mantém motivado a continuar, caso julgue benéfico.
- Relacionamento: se refere a toda interação que pode ocorrer, seja essa com amigos, colegas de equipe e oponentes, sendo de grande valia para o jogo.

As mecânicas de jogos são como as engrenagens de nossa aplicação, esses elementos provocam as ações no jogo e como consequência movem os jogadores (RIBEIRO, 2018). As mecânicas são diretamente responsáveis pelas interações e relacionamentos que os participantes têm com o ambiente do jogo (SCHELL, 2008). Em seu trabalho Werbach e Hunter (2012) descreve algumas mecânicas, sendo essas:

- Desafios: atividades ou tarefas que requerem esforço para serem resolvidas. Tido como um dos elementos mais importantes, pois tornam o jogo interessante para os jogadores.
- Sorte: um item que dê a sensação de aleatoriedade ao jogador, como, por exemplo: dados, cartas ou moedas.
- Cooperação: para atingir o estado de vitória o jogador depende da colaboração de outros integrantes, garantindo o desejo de estar com outras pessoas engajadas em uma atividade.
- Competição: oposto da cooperação. Enquanto a cooperação promove que os jogadores construam algo juntos, a competição faz com que um tente superar o outro. Esses elementos podem gerar o sentimento de vitória ou derrota.
- *Feedback*: ponte de comunicação entre o participante e o jogo. O jogador recebe do jogo respostas, sendo essas positivas ou negativas. O *feedback* permite acompanhar o desempenho e alterar a estratégia para resolver um desafio quando necessário.
- Aquisição de Recursos: em muitos jogos é proposto que para realizar alguma tarefa é preciso realizar um acúmulo de itens.
- Recompensas: são benefícios que o jogador atinge ao concluir desafios, atingir metas ou finalizar o jogo. Pode ser representado por badges (distintivos), vidas extras ou até mesmo direito a jogar novamente.

- Transações: trocas de recursos entre jogadores, seja de forma direta ou por intermediários.
- Turnos: quando dois ou mais jogadores jogam em turnos alternadamente, tal como no “jogo da velha”.
- Estado vencedor: representa a vitória em um jogo, ou seja, o estado final do game. Pode ser retratado por quem atingir o maior número de pontos, conquistar a maioria do território ou eliminar o maior número de concorrentes. Além disso, o estado de vitória deve estar bem definido nas regras.

Componentes de jogos são a parte mais específica dos elementos de jogos, já que representam a essência da dinâmica e mecânica. Além disso, estão ligados diretamente à interface do usuário. Os componentes de jogos são:

- Realizações: mecanismo que recompensa o jogador por cumprir um desafio definido por um objetivo.
- Avatares: representação visual de um jogador usando personagens que mostra ao participante seu papel no sistema gamificado.
- Medalhas (*Badges*): representação visual das conquistas do jogador ao atingir um objetivo. Podem ser simbolizados por selos, insígnias ou outros fatores visuais.
- “Boss fights”: um grande desafio que precisa ser vencido permitindo ao jogador ir para uma próxima etapa.
- Coleções: conjuntos de itens que podem ser coletados e colecionados pelo jogador no decorrer do jogo, podendo ser, por exemplo, peças usadas em desafios futuros ou distintivos que atestam um desafio vencido.
- Combate: uma batalha travada no jogo envolvendo dois ou mais jogadores.
- Desbloqueio de Conteúdos: destravamento de conteúdo ao realizar uma ação, ou seja, o jogador pode desbloquear algum item que antes se encontrava indisponível.
- Doar: oportunidades de compartilhar algum recurso com os demais participantes no sistema gamificado.
- Placar ou *ranking*: permite que o jogador veja sua posição em relação aos demais participantes através de uma lista, podendo ser ordenada pela quantidade de pontos ou itens.
- Níveis: graus que representam a evolução do jogador à medida que avança no sistema gamificado. A ação de subir de nível pode ser disparada ao atingir uma quantidade de pontos ou realizar uma lista de desafios.

- Pontos: forma quantitativa de recompensar os jogadores. Ao realizar uma atividade, o integrante pode receber uma quantidade de pontos definidos pelas regras do jogo.
- Gráfico social: representa a rede de relações dos participantes no sistema gamificado, estendendo o ciclo social do jogador.
- Missões: equipes que trabalham juntas para atingir um objetivo comum.

Para motivar um grupo de pessoas em uma atividade os elementos de jogos podem ser convertidos em atividades com experiências motivadoras. Entretanto, o uso de elementos aleatórios não garante bons resultados (KAPP, 2012). Deste modo, é fundamental compreender cada um dos elementos de jogos, entendendo as formas de funcionamento de cada elemento. Assim, será possível implementar a ludificação em um projeto de modo eficiente (ALVES, 2015).

2.2.2 Categorias de jogadores

Em um jogo cada jogador apresenta um perfil diferente, que é definido pela personalidade e objetivos que o jogador busca alcançar (VIANNA et al., 2013). Deste modo, Bartle (2003) propôs o Modelo de Bartler (Figura 4), que classifica os perfis dos jogadores em quatro categorias, de acordo com o tipo de interesse de cada personalidade. No Modelo, o eixo horizontal representa o interesse do jogador em interagir com outros jogadores ou com o mundo do jogo. Já o eixo vertical indica se o jogador prefere interação ou ação (BARTLE, 2003).

Figura 4 – Grupos de perfis de jogadores

Fonte: (BARTLE, 2003)



O Modelo de Bartle (Figura 4) classifica as diversas classes de comportamento de jogadores em quatro personalidades, sendo elas: (1) predadores, que se divertem agindo contra outros jogadores; (2) conquistadores, que preferem agir no mundo do jogo; (3) socializadores, que optam por interagir com outros jogadores; (4) e exploradores, com interesse em interagir com o mundo do jogo (BARTLE, 2003). A seguir, são definidos cada um dos quatro perfis de jogadores do modelo de Bartle (2003).

- Predadores: jogadores com esse perfil se sentem motivados em obter vitórias baseadas na perda de seus concorrentes. São participantes orgulhosos com suas vitórias individuais, que se sentem atraídos em estar em uma disputa. De certo modo, para o predador o importante é ser o melhor jogador. Elementos de jogos associados: status, desafios e ranking.
- Conquistadores: jogadores interessados em realizar ações no mundo do jogo. Costumam mergulhar no ambiente com foco em dominar o jogo e moldá-lo ao seu gosto. Conquistadores tendem a ser orgulhosos do seu status, valorizam seus placares e prezam pelas próprias conquistas. Elementos de jogos associados: medalhas, níveis e recompensas.
- Exploradores: são interessados em explorar todo o mundo do jogo e assim desvendar todas as possibilidades. Para esse perfil o importante é a trajetória. Além disso, esses jogadores tendem a ter orgulho do seu nível de conhecimento. Exploradores Também valorizam o reconhecimento de outros jogadores sobre seu conhecimento sobre o mundo do jogo. Elementos de jogos associados: missões, conteúdos desbloqueados e recompensas.
- Socializadores: jogadores que buscam interação com os demais, onde socializar com outros é mais importante que atingir o objetivo. Preferem jogos que envolvam cooperatividade e vínculos sociais, além de valorizar suas amizades e suas influências no mundo do jogo. Elementos de jogos associados: missões, gráfico social e equipes.

É comum que o jogador tenha sua personalidade associada às características de um perfil. Entretanto, também é possível que sua personalidade seja associada a inúmeros perfis diferentes. Desta forma, cada jogador pode apresentar um tipo diferente de diversão ao realizar uma atividade (SILVA; BAX, 2017). Em vista disso, Lazzaro (2004) classificou as diversões, sendo essas:

- *Hard fun*: associado a resolver problemas e a superação de obstáculos.
- *Easy fun*: diversão relacionada a explorar e se divertir.
- *People fun*: ligada ao trabalho em grupo e à socialização.

- *Serious fun*: é associado em realizar algo que seja significativa para a sociedade.

Assim como cada jogador apresenta um perfil diferente, a diversão manifestada também difere entre os participantes em um ambiente gamificado. Deste modo, a diversão deve ser considerada ao desenvolver uma solução ludificada. Buscando simplificar essa questão, Silva e Bax (2017) realizou uma associação dos perfis com as respectivas categorias de diversão de um ambiente gamificado no contexto educacional (Tabela 1). Além disso, podemos observar também as emoções ligadas a cada perfil.

Tabela 1 – Associação entre os perfis de estudantes, tipos de diversão e emoções.

Fonte: (SILVA; BAX, 2017, pg. 155)

Perfil	Tipos de diversão	Emoções associadas
Conquistadores: motivados pela realização das seguintes atividades: assistir às aulas, resolver os desafios em forma de exercícios, finalizar os tópicos	<i>Hard fun</i>	Satisfação, excitação, frustração, foco, relaxamento
Exploradores: motivados mais pelo conteúdo do que pelos desafios.	<i>Easy fun</i>	Curiosidade, surpresa, temor
Socializadores: motivados principalmente pelas mecânicas que promovem interação e cooperação entre estudantes.	<i>People fun</i>	Divertimento, amizade, admiração, intimidação, rejeição
Predadores: incentivados à competição por meio de vitórias, reconhecimentos e comparações de status.	<i>Hard fun e Serious fun</i>	Triunfo, excitação, alívio, frustração, foco

Todo jogador apresenta um sentimento diferente com relação à diversão de um jogo. Em outras palavras, o que é divertido para um jogador pode não ser interessante para outros participantes. Em vista disso, o modelo de uma plataforma gamificada deve ser planejada de modo a atender os diferentes perfis e suas respectivas emoções /citeEN-CONTROSBIBLI1518.

2.3 Aplicações da gamificação no contexto educacional

A gamificação no contexto educacional é algo relativamente novo. Entretanto, já existem plataformas que demonstraram sucesso em motivar alunos usando de elementos de jogos. Objetivos claros e feedbacks são exemplos de elementos que causam um efeito positivo na motivação dos alunos. Em outras palavras, os participantes de experiências gamificadas se sentem mais motivados com metodologias que promovem a proatividade dos alunos (BRITO, 2017).

Um exemplo de aplicação da gamificação no contexto educacional foi aplicada pelo professor Sheldon (2011), que gamificou a disciplina sobre *Game Design* no Instituto Politécnico Rensselaer, em Nova Iorque, nos Estados Unidos. Para isso, Sheldon (2011) transformou a disciplina em um jogo, no qual trabalhos em grupos foram convertidos em missões, notas em pontos de experiência (*Experience Points*) e alunos em avatares. O professor Sheldon obteve resultados positivos, além de destacar que para aplicar a gamificação não é preciso mudar os objetivos de ensino, já que o método aplicado é apenas um dos caminhos para atingir o aprendizado (FARDO, 2014).

Figura 5 – Capturas da ferramenta Duolingo

Fonte: Capturado pelo autor

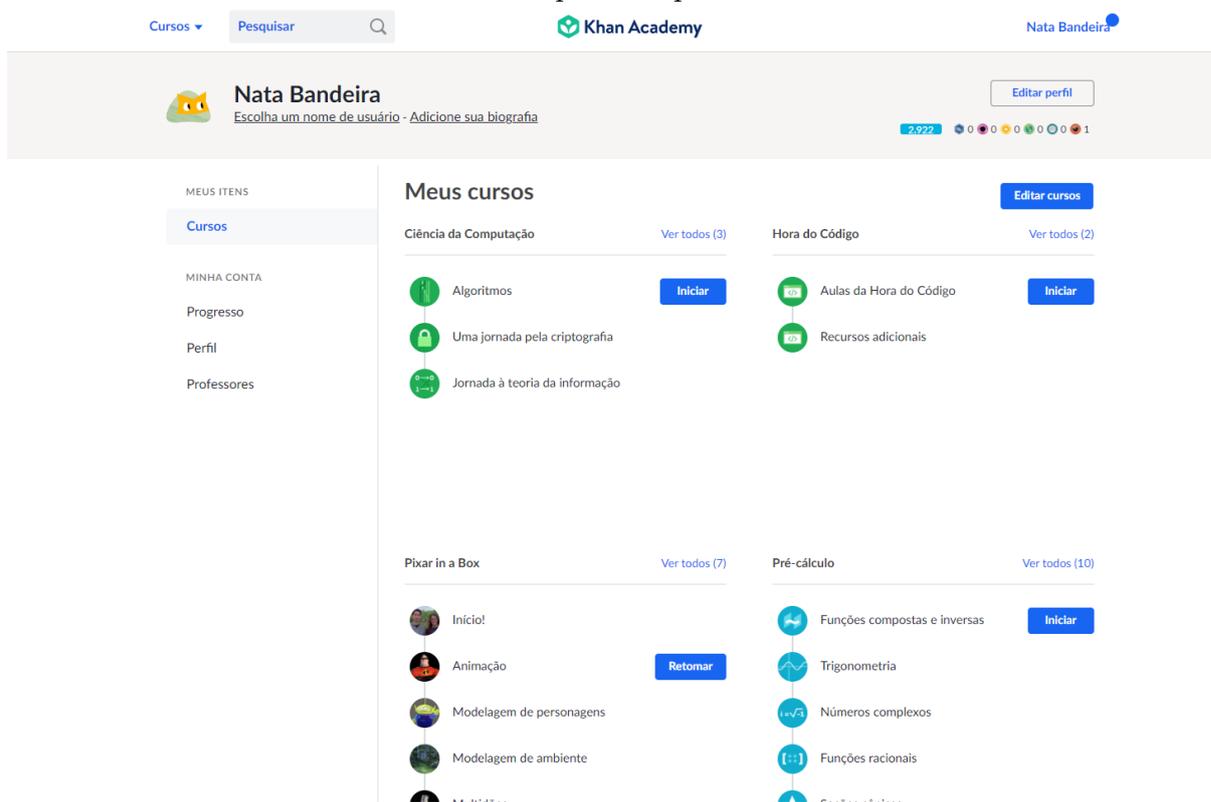


Outro exemplo de aplicação é o Duolingo (Figura 5), que é um aplicativo para IOS e Android voltado ao ensino de linguagens gratuitamente. O Duolingo usa de alguns elementos gamificados. São eles: placares, desbloqueio de conteúdo, níveis, *badges* (dis-

tintivos) e realizações. Além disso, o jogo cria metas diárias que mantêm o usuário focado e motivado em realizar as missões. Graças a isso, a ferramenta é considerada um caso de sucesso de gamificação, já que acumula milhões de downloads nas lojas de aplicativos do IOS e do Android (DUOLINGO, 2021).

Figura 6 – Capturas da ferramenta Klan Academy

Fonte: Capturado pelo autor



Klan Academy, que é exibida na Figura 6 é uma ferramenta online de ensino e aprendizado que pode ser apontada como outro exemplo de aplicação da gamificação na educação. Criada em 2008, a Klan Academy foi desenvolvida com o propósito de prover educação gratuita para todas as pessoas. Além disso, a ferramenta usa de gamificação para motivar os alunos em suas atividades. Por exemplo, é possível observar o uso de medalhas em caso de algumas conquistas realizadas pelos alunos. Outro elemento de jogo aplicado são os avatares, que são personagens que representam os alunos e que evoluem à medida que ocorre o progresso nos estudos (KHAN... , 2021).

Hess e Gunter (2013) desenvolveram um trabalho com o intuito de comparar a experiência dos alunos em aprendizados baseados em jogos contra não baseados em jogos. Buscando analisar o tópico, foi realizado uma experiência com dois cursos online sobre história americana. Um dos cursos era *game-based* (Baseado em jogos) enquanto o outro era *nomgame-based* (Não baseado em jogos). O experimento buscava analisar quatro pontos comparando os dois cursos. São eles:

- A comparação do tempo que alunos levam para concluir cada um dos cursos;

- Com a aplicação de um sistema baseado em jogos ocorre um aumento da performance dos alunos;
- Qual relação que o aluno desenvolve com cada curso e a motivação intrínseca de cada participante;
- Quais aspectos de cada um dos cursos (*game-based* e *nomgame-based*) que os alunos consideraram úteis ou desvantajosos ao aprendizado.

Os resultados da pesquisa demonstraram que os alunos do curso *game-based* levaram um período maior de tempo para realizar as atividades. Entretanto, os alunos do curso baseado em jogos apresentaram melhores resultados médios. Outro ponto destacado é que alunos do curso *game-based* apresentaram mais interações com o curso. Por fim, os alunos e professores do curso baseado em jogos apontaram como pontos positivos o formato novo de aprendizado, as avaliações, gráficos e vídeos online. Como pontos negativos, foram apontados problemas técnicos e a voz dos personagens do jogo.

3 METODOLOGIA

As atividades práticas propostas pelos professores em sala podem auxiliar na fixação do conteúdo. Entretanto, muitos alunos apresentam baixo engajamento para realizar exercícios. Deste modo, este trabalho tem como foco desenvolver o protótipo de uma ferramenta gamificada Learn, Play and Code(LPC), que é voltada à aplicação de atividades na disciplina de Introdução à Programação.

Para implementar o protótipo do LPC, foi utilizado o Modelo de Aplicação de Atividades Gamificadas de Introdução à Programação (MAAGIP). Deste modo, o MAAGIP define como a gamificação é adotada no contexto da aplicação de atividades de IP. Para implementar o modelo gamificado é necessário estabelecer os Objetivos Gamificados, já que uma solução sem propósitos claros pode gerar resultados não significativos (BOSSE; GEROSA, 2015).

3.1 Objetivos gamificados

Todo projeto gamificado deve ter um objetivo claro do que se pretende com a aplicação dos elementos de jogos. Os Objetivos Gamificados definem qual o resultado esperado na implementação de uma ferramenta ludificada. Em vista disso, a ferramenta LPC tem como objetivos gamificados:

1. Tornar o processo de praticar programação mais interessante e atrativo para os alunos;
2. Manter os alunos engajados ao realizarem atividades da disciplina, tanto em sala de aula como em estudo extraclasse;
3. Promover aos alunos um aprendizado mais “divertido” dos conceitos de programação;
4. Por fim, melhorar a taxa de aprendizado dos alunos.

Para definir se o LPC atinge os objetivos gamificados, deve-se analisar se houve melhoria no processo de aprendizado dos alunos, além de verificar se ocorreu uma redução no percentual da evasão de alunos na disciplina de IP. Entretanto, essa parte de análise não faz parte deste trabalho, e fica como trabalho futuro o processo de validar e evoluir a ferramenta.

3.2 Implantação do modelo MAGIP

O Modelo de Aplicação de Atividades Gamificadas de Introdução à Programação (MAAGIP), foi definido para estruturar como os elementos de jogos serão aplicados no

contexto da aplicação de atividades de IP. Deste modo, o MAAGIP define: (1) quem são os jogadores e como interagem com a ferramenta; (2) o cenário, que é o ambiente onde os elementos gamificados estão dispostos; (3) os elementos de jogos escolhidos para gamificar a ferramenta; (4) e por fim, as regras do ambiente gamificado.

O primeiro elemento definido do modelo é o perfil dos jogadores, que é o público-alvo da ferramenta LPC, ou seja, os participantes que interagem com os elementos de jogos. O MAAGIP é direcionado a alunos que estejam cursando a disciplina de IP em cursos de computação, ou que contenham ensino introdutório de programação. Também é necessário que os alunos estejam na mesma turma, visto que cada turma tem um ambiente gamificado individual conduzido pelo professor da disciplina, que define as atividades que serão aplicadas.

Já o cenário é o ambiente no qual os participantes interagem com a ferramenta gamificada. O MAAGIP foi pensado para ser usado na aplicação de atividades em sala de aula ou em extraclasses. Em ambos os casos, os alunos precisam de computadores que tenham acesso à Internet, já que para visualizar as atividades propostas e seus resultados, os alunos devem acessar uma página Web. Também é preciso que tenha instalado nos computadores o editor de código VSCode com a *extension* da ferramenta LPC, que permite resolver as atividades e enviar as respostas à LPC.

Com o cenário e os jogadores estabelecidos, foi determinado os elementos de jogos aplicados no modelo. O MAAGIP adota os seguintes elementos:

1. Pontos, que são formas quantitativas de contabilizar o acerto dos alunos ao realizarem atividades;
2. *Avatar* que representa o aluno como um jogador no ambiente da ferramenta LPC;
3. *Badges*, que também podem ser chamados distintivos, ou seja, emblemas que indicam conquistas realizadas pelos alunos;
4. Fases representam os níveis dos jogos e provem ao aluno a sensação de evolução à medida que realizam atividades;
5. Barra de progresso, que permite ao aluno visualizar sua evolução em uma fase;
6. *Ranking* classifica os alunos por suas pontuações em uma lista, onde as melhores pontuações são posicionadas no início da lista.

Cada um dos elementos de jogos definidos tem uma funcionalidade específica, por exemplo, a escolha de pontos, *rankings* e *badges*, buscam desenvolver competitividade entre alunos. Já as fases e a barra de progresso permitem que os alunos acompanhem suas evoluções à medida que realizam as atividades. Ainda assim, os elementos de jogos tem comportamentos e funcionamentos definidos nas regras do modelo da ferramenta.

No MAAGIP, as regras definem como cada elemento de jogo se comporta, delimitando: (1) a quantidade de pontos que um aluno recebe e em que casos ocorre a pontuação; (2) quando ocorre a conquista de um *badge*; (3) como é realizada a classificação dos alunos nos *rankings*; (4) o tempo para realizar uma fase; (5) e por fim, os tipos de pontuações.

3.3 Prototipar o modelo gamificado

O processo de prototipar o modelo gamificado consiste em propor e construir uma ferramenta de software que use dos elementos de jogos na realização das atividades de IP. Deste modo, é proposto o LPC (Learn, Play and Code), que é uma ferramenta que usa do modelo MAAGIP. O LPC tem sua estrutura baseada na arquitetura de microsserviços, já que esta é dividida em quatro serviços. São eles: autenticação, repositório de atividades, gamificação e compilação.

O Serviço de Autenticação (LPC-Auth) é então desenvolvido com o intuito de validar o acesso dos usuários à ferramenta, delimitando também o nível de acesso. Ou seja, usuários do tipo professor conseguem administrar turmas enquanto os alunos podem somente visualizar a parte gamificada da ferramenta com sua pontuação, fases, *ranking* e *feedbacks*.

O Repositório de Atividades (LPC-Repository) armazena e cataloga todas as atividades da ferramenta. Deste modo, no momento em que os professores estiverem criando fases para suas turmas, será possível buscar atividades no repositório para enviar aos alunos. É possível também que um professor proponha novas atividades e as compartilhe com outros educadores através do LPC-Repository.

O Serviço de Gamificação (LPC-Game) foi estruturado para realizar o controle dos elementos de jogos. Deste modo, o LPC-Game aplica as pontuações em casos de acerto de atividades, gerencia as fases de cada turma, envia *feedback* após um aluno resolver atividades e atualiza o ranking de alunos. Para compilar as atividades, o serviço de compilação(Judge0) é adotado como sistema de execução de código.

Para comunicarem-se, os serviços do LPC usam de APIs (Applications Protocol Interface) para enviar solicitações. O serviço que recebe a solicitação pode responder com informações requeridas, ou até mesmo realizar alguma operação, como atualizar alguma instância no banco de dados.

3.4 Ferramentas usadas

O LPC usa diversas ferramentas para poder gamificar a aplicação de atividades de Introdução à Programação. Essas ferramentas são bibliotecas de software para armazenamento de dados, sistema de backend, implementação de *front-end* (página web). A seguir são listadas as ferramentas escolhidas para se estruturar o LPC.

- Flutter
- VsCode
- ExpressJS
- PostgreSQL
- Prisma
- Judge0

O framework Flutter foi escolhido para implementação das interfaces gamificadas dos alunos e da página que os professores usarão para administrar as fases de suas turmas. Desenvolvido pela Google, Flutter é um *framework* de código aberto usado para desenvolver aplicações *mobile*, *web* e *desktop*. Para resolver atividades, os alunos usarão o editor de código VSCode (Visual Studio Code) com a extensão (*extentions*) LPC-Code. O VSCode foi desenvolvido pela Microsoft, e possui suporte para que terceiros desenvolvam extensões de funcionalidades para o editor. Graças a isso, o LPC-Code foi idealizado para que os alunos possam resolver as atividades usando o VSCode, já que a ferramenta já é um editor de código bem estruturado e conhecido por muitos desenvolvedores de software.

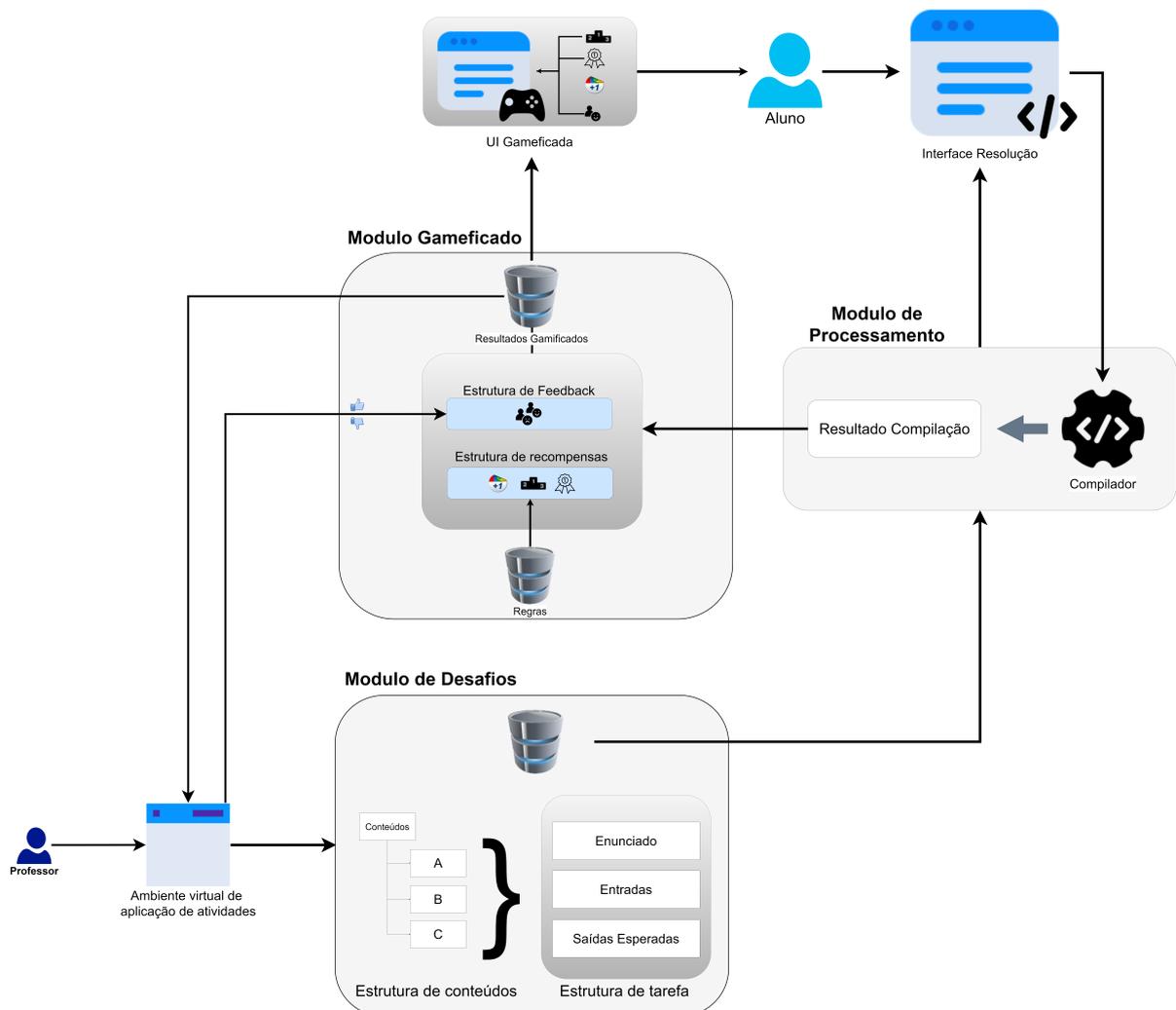
Para implementação dos serviços do LPC é usado: (1) PostgreSQL, para banco de dados; (2) Prisma como um ORM; (3) e ExpressJS para o servidor. PostgreSQL é um gerenciador de banco de dados relacional desenvolvido com uma aplicação de código aberto. Já o ExpressJS é *framework* para NodeJs usado para construção de servidores para web. Por fim, o Prisma é um ORM (*Object Relational Mapper*) usado para realizar consultas no banco de dados.

O Judge0 é então escolhido como ferramenta de compilação do LPC, ou seja, é usado para executar os códigos desenvolvidos pelos alunos como soluções para as atividades. O Judge0 é um executor de código *open-source* que pode ser usado em plataformas de aprendizado de programação, editores de códigos online, IDEs online, etc.

4 MODELO DE APLICAÇÃO DE ATIVIDADES GAMIFICADAS DE INTRODUÇÃO A PROGRAMAÇÃO (MAAGIP)

Como é exibido na Figura 7, o Modelo de Aplicação de Atividades Gamificadas de Introdução a Programação (MAAGIP), é um modelo de uma ferramenta gamificada proposta por esse trabalho para a aplicação de atividades da disciplina de Introdução a Programação para ludificar o processo de aprendizagem dos alunos. O MAAGIP dispõe de três Módulos principais, são eles: Módulo de Desafios (MD), responsável por tratar e organizar as atividades propostas pelos professores; Módulo de Processamento (MP), que compila e executa as soluções das atividades realizadas pelos alunos; e, por fim, o Módulo de Gamificação (MG), que avalia as respostas dos alunos e converte os resultados em uma estrutura ludificada e a disponibiliza aos participantes (professores e alunos).

Figura 7 – Estrutura do MAAGIP

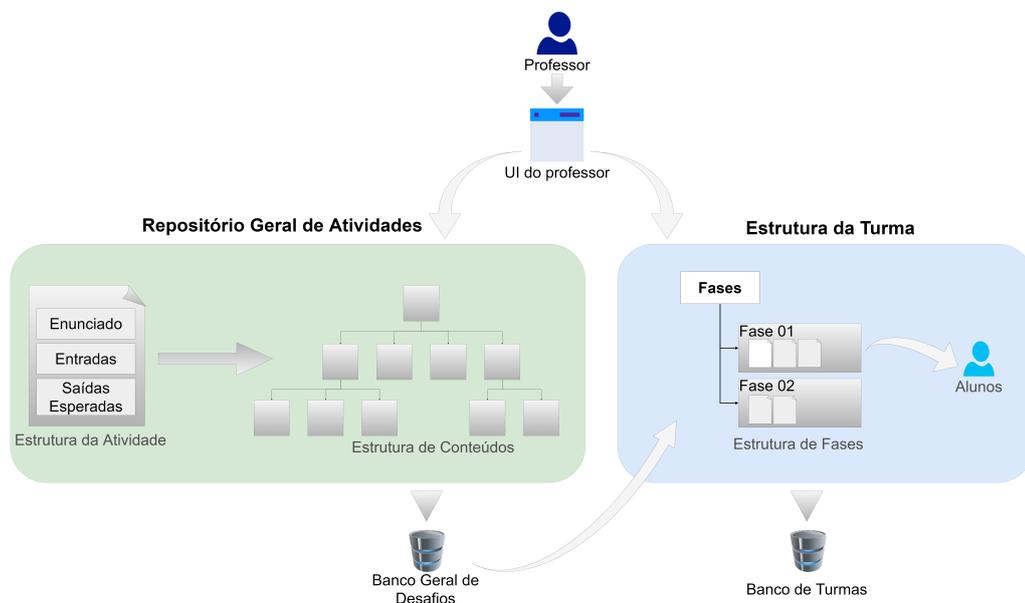


Na estrutura do modelo (Figura 7), também são apresentados os participantes (Professores e Alunos) e os elementos auxiliares de interface, que estabelecem uma relação entre os participantes e os Módulos do modelo. O MAAGIP usa três elementos de interfaces: (1) o Ambiente Virtual de Aplicação de Atividades (AVAA), que possibilita ao professor propor desafios e fornecer feedbacks nas resoluções dos alunos; (2) Interface de Resolução, que é um editor de código em que o aluno resolve os desafios propostos nas atividades; (3) e, por fim, UI Gamificada, que usa de uma linguagem ludificada para representar os resultados finais.

4.1 Módulo de Desafios

O Módulo de Desafios (MD) realiza o processamento e armazenamento das atividades propostas pelo professor através da UI do Professor (UIP). O MD dispõe de quatro elementos, são eles: (1) O Repositório Geral de Atividades (RGA) estrutura as atividades (desafios) existentes na ferramenta através de uma Estrutura de Conteúdos; (2) Estrutura da Turma (ET), que organiza as atividades de uma turma usando uma Estrutura de Fases, e relaciona um grupo de alunos a uma ET; (3) Banco de Desafios Geral, que armazena as atividades do RGA; (4) Banco de Turmas, que armazena as atividades aplicadas em uma turma e o registro dos alunos.

Figura 8 – Módulo de Desafios



Neste módulo, a Estrutura da Atividade (Figura 8) proposta pelo professor representa as partes do desafio, que são, Enunciado, Entradas e Saídas Esperadas. O Enunciado é a descrição do problema a ser resolvido pelo aluno. Já as Entradas, são valores usados

como parâmetros para execução da resolução do exercício. Por fim, as Saídas Esperadas são valores finais esperados como resultado da execução da solução do desafio desenvolvido pelo aluno.

A Estrutura de Conteúdos (EC)(Figura 8) representa como os tópicos e sub-tópicos da ementa da disciplina de Introdução a Programação (IP) são organizados de modo hierárquico. Cada item (Tópico) da EC equivale a um conteúdo de IP, que está relacionado a um conjunto de atividades. Na estrutura hierárquica os tópicos podem ou não ter subtópicos anexados conforme a ementa de IP. Portanto, os professores que interagem com o repositório de atividades podem propor novos desafios para um tópico, ou realizar pesquisa de desafios através dos itens da Estrutura de Conteúdos.

O Repositório Geral de Atividades (RGA)(Figura 8) é um arquivo que matem uma coleção de desafios. Todas as atividades propostas pelo professor são estruturadas pela Estrutura de Conteúdos e armazenadas no Banco Geral de Desafios. Por isso, através da UI do Professor, é possível compartilhar atividades com outros educadores, ou buscar desafios para aplicar na sua própria turma, criando assim uma rede de reaproveitamento de atividades.

A Estrutura da Turma (ET)(Figura 8) equivale a uma classe de aula, composta por: um professor, que supervisiona a turma e propõe atividades do RGA; um grupo de alunos, que serão o alvo da estrutura gamificada e realizarão as atividades; e a Estrutura de Fases, que representa como os desafios são organizados em etapas. Cada fase simboliza um conjunto de atividades propostas pelo professor. Com intuito de ter um modelo para uma ferramenta flexível à metodologia de ensino da disciplina, o professor tem a liberdade de propor as fases onde pode determinar: quantidade de fases; as atividades que compõem cada fase; e por fim, o tempo para conclusão de uma fase.

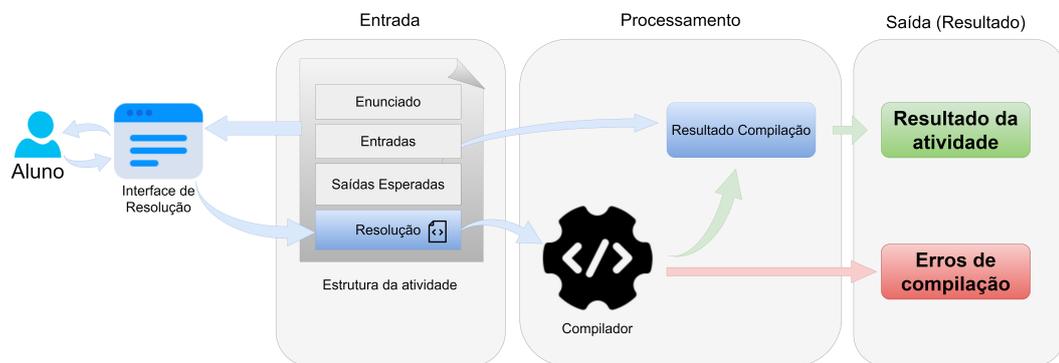
O Modulo de Desafios usa duas estruturas de banco de dados, são elas, Banco Geral de Desafios (BGD) e Banco da Turma (BT) (Figura 8). O BGD armazena as atividades estruturadas pela Estrutura de Conteúdos. Já o BT, armazena os dados de cada turma relacionando professor, alunos e desafios organizados pela Estrutura de Fases. De modo a evitar inconsistências como tarefas com título e enunciado alterado após a aplicação, uma cópia da Estrutura de Atividades é transferida do Repositório Geral de Atividades para a Estrutura da Turma, ou seja, do BGD para o BT. Deste modo, os alunos não perceberão alterações, se uma atividade do BGD for alterada após ser aplicada em uma turma.

Com a definição do Repositório Geral de Atividades e Estrutura da turma no Módulo de Desafios, se inicia a próxima etapa do MAAGIP, que é o Modulo de Processamento. Neste sentido, o Módulo de Processamento (Seção 4.2) compila os resultados das atividades resolvidas pelos alunos, e disponibiliza os resultados de compilação para a próximas etapas do modelo.

4.2 Módulo de Processamento

O Módulo de Processamento (MP), como exibido na Figura 9, compila e executa as respostas dos desafios realizados pelos alunos. A sua estrutura é dividida em três etapas, são elas: Entrada, que é a estrutura da atividade com a resolução do desafio; Processamento, que compila e executa a resolução da atividade; e Saída, que dispõe para outros Módulos os resultados da execução da atividade ou os erros de compilação.

Figura 9 – Módulo de Processamento



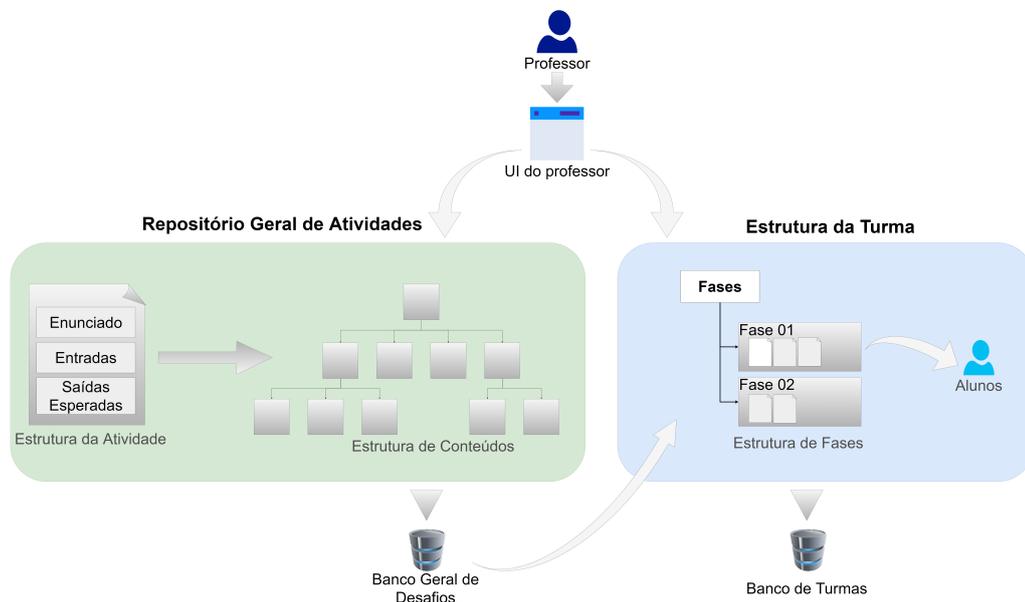
A Entrada do MP consiste em uma atividade da Estrutura de Atividades do Módulo de Desafios. A atividade é resolvida pelo aluno através da Interface de Resolução (IR), assim, a EA tem adicionada a sua estrutura a Resolução, ou seja, o código-fonte desenvolvido pelo aluno como resposta à atividade. Na fase de Processamento ocorre a compilação e execução da Resolução, que pode ocorrer em linguagens compiladas ou interpretadas. No caso de uma linguagem compilada o código-fonte é convertido em um executável e executado a posteriori. Já em linguagens interpretadas como scripts, a etapa de compilação é ignorada já que o código-fonte é executado diretamente por um interpretador.

O Resultado da Compilação usa as Entradas da Estrutura da Atividade (Figura 9) como parâmetros para executar os códigos de resolução, o que resulta na Saída. A Saída do processamento tem dois retornos possíveis, são eles: os Erros de Compilação ou Resultados da Atividade. Os Erros de Compilação são disparados em caso de inconsistência no código-fonte implementado pelo aluno, seja por razão léxica, semântica ou sintática. Já os resultados da atividade são as saídas da execução da resolução do desafio quando não há erros de compilação. Por fim, o Resultado da Atividade é disponibilizado para ser analisado pelo Módulo de Gamificado, que determina se a Saídas Esperadas são válidas.

4.3 Módulo Gamificado

O Módulo Gamificado (MG) (Figura 10) trata os resultados das atividades fornecidas pelo Módulo de Processamento. No MG a solução das atividades são convertidas em uma estrutura ludificada. A sua estrutura é composta por: Estrutura da Atividade, estendida com as saídas do MP; Resultados Gamificados, que ludifica os resultados das atividades; Banco de Regras, que contém as diretrizes a serem seguidas pelo MG; Resultados de Feedback, que são análises fornecidas por professores com relação à resolução de uma atividade; e Banco de Resultados Gamificados que armazena os resultados do MG.

Figura 10 – Módulo Gamificado



Como exibido na Figura 10, a EA tem atribuída na sua estrutura os erros de compilação ou resultados da atividade, que são providas pelo Módulo de Processamento. As saídas do MP são utilizadas para gerar os resultados gamificados através de uma estrutura de pontos. Desta forma, os acertos do aluno são convertidos em pontos positivos e os erros em pontos negativos. Um ponto negativo pode ser computado de duas formas: por erro de compilação, que indica que o aluno cometeu uma falha na implementação do código-fonte; ou quando as Entradas fornecidas pela EA não condizem com as Saídas Esperadas. Já os pontos positivos, ocorrem quando não há erros de compilação e as Entradas condizem com as Saídas Esperadas.

Os Resultados Gamificados (Figura 10) moderam os elementos de jogos dispostos para os participantes. Desta forma, itens como *badges* (medalhas) e *ranking* de posição são constantemente atualizados à medida que os pontos são alterados. Para isso, os resultados gamificados usam de diretrizes armazenadas no Banco de Regras (Figura 10).

Essas regras especificam: a quantidade de pontos que um aluno recebe, que depende de quantas tentativas foram realizadas; e perda de pontos, que pode ocorrer como punição em casos de conduta inadequada do aluno, como a não realização das atividades no período estipulado.

O Resultado de *Feedback* (Figura 10) é fornecido pelo professor como uma resposta à resolução das atividades realizadas pelos alunos. O *feedback* é opcional ao professor, e serve para que a resolução do aluno tenha uma análise mais detalhada que a fornecida pelo sistema gamificado. Deste modo, o estudante tem uma melhor visão de seus erros, podendo assim corrigi-los com mais facilidade. Por fim, todos os resultados do Módulo de gamificação são armazenados no Banco de Resultados Gamificados para serem consumidos por alunos e professores.

4.4 Elementos da Ferramenta Gamificada.

Os elementos da ferramenta representam as partes de uma aplicação gamificada, são eles: participantes (Professores e Alunos), que são as personas que interagem com a ferramenta; Cenário, que é o ambiente onde os participantes interagem para realizar suas funções; desafios, que são as atividades realizadas pelos alunos; elementos de jogos, usados para engajar alunos em suas atividades; regras, que determinam limitações e funções da ferramenta e dos alunos; e modelo de engajamento dos alunos, que determina como ocorre a interação dos participantes com a solução gamificada;

4.4.1 Cenário (Ambiente)

Cenário ou ambientes de jogos, é o espaço disposto aos jogadores (alunos) para que esses se envolvam com as atividades propostas. O MAAGIP tem como cenário um ambiente virtual (UI do aluno) que dispõe de perfil dos jogadores, mapa de progresso, *ranking*, *badges* (distintivos), página de atividades e página de *feedback*. A seguir, cada um dos itens listados são especificados.

- Perfil dos jogadores: dispõe informações sobre o aluno, tais como nome, conquistas, pontos e imagem do avatar (representação do aluno). Permite também que um jogador visualize o perfil de seus colegas. Também são exibidos nessa página os *Badges* (distintivos), que são representações visuais de uma conquista ou realização.
- Mapa de progresso: representação visual que exhibe o progresso do aluno à medida que realiza as fases. Deixando destacado ao participante a fase atual e as futuras que precisam ser resolvidas.
- *Ranking*: listagem que usa a quantidade de pontos para organizar os alunos em ordem crescente usando os pontos. Demonstrando para o aluno sua posição no ranking em relação aos seus colegas de classe.

- Página de atividades: listagem dos desafios propostos pelo professor em formato de quests (Fases).
- Interface de resolução: página que o aluno resolve uma atividade, assim, tem disposto na interface a descrição do desafio, as entradas, saídas esperadas e um editor de código usado para postar a resolução do desafio.
- Página de *feedback*: exibe as avaliações providas pelo professor com relação aos resultados das resoluções dos desafios.

Assim como os cenários representam os ambientes de interação dos usuários com o modelo da ferramenta gamificada, há também outro elemento de jogo crucial que são os jogadores que interagem com o ambiente. Portanto, no próximo tópico é descrito os participantes e suas características.

4.4.2 Jogadores (Participantes)

Os participantes da aplicação representam as classes de usuários que interagem com solução gamificada. O MAAGIP tem duas categorias de participantes, são eles, professores e alunos. O professor é o mentor da ferramenta gamificada, e tem as seguintes atribuições:

- Criar turmas e permitir que alunos se inscrevam através de um sistema de convite enviado por um link;
- Propor novas atividades para Repositório Geral de Atividades, podendo assim compartilhá-las com outros educadores;
- Propor novas fases para uma turma e determinar as atividades de cada fase conforme a metodologia de ensino do professor;
- Prover feedbacks na resolução de desafios, gerando pontos positivos ou negativos para o sistema de placar do aluno;
- Monitorar o andamento das atividades, podendo punir um aluno com redução de pontos em caso de conduta inadequada do mesmo, como trapaças.

Os alunos (jogadores) são os participantes alvo da solução gamificada, visto que estarão em contato com elementos de jogos. Deste modo, podem realizar as seguintes ações no ambiente ludificado:

Se inscrever em uma nova turma. Passar de fases e realizar as atividades propostas pelo professor. Interagir com outros participantes, seja indiretamente se competem com outros por posições melhores dentro ranking, ou de forma direta, ajudando os demais com

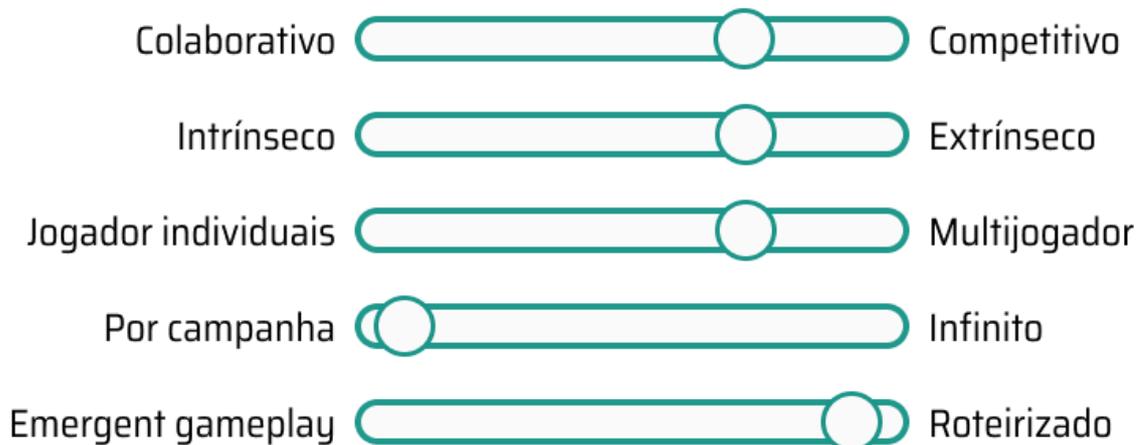
suas dificuldades com o conteúdo. Acompanhar a evolução própria ou dos colegas através da interfaces de perfis. Receber feedback em formato de pontos, rankings ou realizado diretamente pelo professor.

Após definir as categorias de jogadores que interagem com o modelo da ferramenta gamificada é definido o modelo de engajamento dos participantes, que define como os alunos, ou seja, jogadores interagem com o ambiente da ferramenta gamificada.

4.4.3 Modelo de engajamento dos jogadores

O Modelo de engajamento dos jogadores determina como os alunos interagem com a ferramenta gamificada e com outros alunos, estipulando também as recompensas dos participantes (BURKE, 2015). Cada par de parâmetros (Figura 11) especifica uma característica da ferramenta, e a escolha entre ambos estipula como a interação dos alunos com a ferramenta é realizada. A Figura 11, determina os parâmetros de engajamento da solução e o peso entre cada par, são eles: colaborativo e competitivo; intrínseco e extrínseco; multijogador e jogador individual; por campanha e infinito; emergent gameplay e roteirizado.

Figura 11 – Parâmetros de engajamento do MAAGIP



- Colaborativo/Competitivo: na ferramenta os alunos competem por posições de rankings e medalhas que podem ser conquistadas se superarem seus colegas. Deste modo o modelo de engajamento é mais competitivo, entretanto, a colaboração entre os alunos também será fomentada através da comunicação entre os participantes.
- Intrínseco/Extrínseco: se refere às recompensas recebidas pelos participantes. O MAAGIP usa elementos extrínsecos, já que os elementos de jogos como pontos e ranking são fatores externos. Entretanto, pode ocorrer do aluno relacionar a sensa-

ção de conquista e crescimento a com a realização das atividades de programação. Deste modo, o aluno pode sentir uma motivação intrínseca ao realizar as atividades.

- Multijogador/Jogadores individuais: majoritariamente a ferramenta é voltada a multijogadores. Na maioria das vezes, a interação dos alunos com os demais participantes ocorre indiretamente. Ou seja, através da competição com outros alunos por pontos, badges e posições no ranking.
- Por campanha/Infinito: o uso da ferramenta gamificada é temporária, e tem como tempo de duração o período que ocorrer a disciplina de IP. Deste modo, a ferramenta é organizada por campanhas, em outras palavras, cada campanha indica um conjunto de atividades (fase).
- *Emergent Gameplay*/Roteirizado: neste caso, a solução se encaixa mais na categoria roteirizada, já que os resultados dos desafios são pré-definidos. Contrariamente ao *Emergent Gameplay*, que se encaixa em casos com situações inesperadas, o que é oposto ao contexto da ferramenta.

O processo de determinar o modelo de engajamento dos jogadores implica na forma como a experiência dos usuários será estruturada, onde cada parâmetro exerce um forte efeito sobre o funcionamento do modelo gamificado (BURKE, 2015). Deste modo, os parâmetros definem como os elementos de jogos vão atuar dentro do MAAGIP.

4.4.4 Elementos de Jogos

O sistema de jogo aplica cinco princípios de game design que foram escolhidos de acordo com uma análise baseada no trabalho de Dicheva et al. (2015). São estes: status visível, progresso, objetivos, competição e colaboração. Cada um dos princípios citados tem um objetivo a ser atingido e podem ou não empregar componentes de jogos. A Tabela 2, mostra cada um dos princípios aplicados no modelo da ferramenta e os componentes de jogos escolhidos.

O uso dos princípios de *game design* são usados como uma abordagem para melhorar o engajamento dos alunos Dicheva et al. (2015). Cada princípio de game design se relaciona a um ou mais componentes de jogos, são eles: pontos e *badges*, que representam emblemas de conquistas exibidas na tela de perfil do aluno; níveis, que permitem aos jogadores acompanharem o seu progresso e visualizar em que ponto do jogo se encontram; e por fim, o *ranking* que usa dos pontos para destacar os alunos com maior pontuação.

Tabela 2 – Associação entre os princípios de game design, objetivos e componente de jogos.

Fonte: Adaptado de Dicheva et al. (2015)

Princípios de <i>game design</i>	Objetivos	Componente de jogo
Status visível	Reputação, credibilidade e reconhecimento	Pontos, <i>avatar</i> (perfil do aluno) e <i>badges</i> (Distintivos)
Progresso	Progresso visível da evolução	Pontos, barra de progresso e níveis
Objetivos	Objetivos específicos claros e imediatos	
<i>Feedback</i>	<i>Feedbacks</i> imediatos ou em curto período ao no lugar de recompensas após grandes ciclos de tempo	
Competição e colaboração		<i>Ranking</i> , <i>avatar</i> (perfil do aluno) e pontos

4.4.4.1 Níveis

Dentro do MAAGIP, na perspectiva do aluno, cada nível equivale a uma fase na estrutura gamificada, assemelhando-se a um jogo. Cada fase é composta por um grupo de desafios, ou seja, tarefas, assim como é descrito no Módulo de Desafios. Além disso, cada fase pode representar um ou mais tópicos da matéria de Introdução a Programação. Os níveis visam proporcionar ao aluno a sensação de progresso à medida que resolvem as atividades de programação.

O professor da matéria de Introdução a Programação tem controle sobre as fases da ferramenta gamificada, e pode escolher como organizar os níveis conforme a metodologia de sua aula. Outro fator que os professores têm controle é o tempo disponível para que os alunos finalizem a fase. Em cada aula o professor pode liberar uma nova fase com um conjunto de atividades que se referem ao conteúdo ensinado em sala de aula. Deste modo, o professor tem a liberdade de propor novos desafio de programação ou buscar no repositório geral de atividades.

Tabela 3 – Descrição do *status* das fases

Status da fase	Descrição
Em Espera	Indica uma fase que está indisponível para os alunos.
Liberada	Fase liberada para os alunos resolverem as atividades.
Iniciada	Fase que aluno iniciou a resolução das atividades, mas ainda não finalizou.
Concluída	Fase na qual o aluno realizou todas as atividades no período delimitado pelo professor.
Não Concluída	Ocorre quando o tempo para realizar uma fase termina e o aluno não iniciou ou não finalizou as atividades.

Cada fase tem um status que indica o estado que fase se encontra (Tabela 3), são eles: Em Espera, Liberada, Iniciada, Concluída e Não Concluída. Uma fase que acabou de ser criada inicia no status Em Espera. A partir do momento que o professor disponibiliza as atividades para a turma o status é alterado para Liberada. Caso o professor queira cancelar a fase ou alterar alguma informação ele pode alterar uma fase Liberada para Em Espera. Cada fase gera uma pontuação que será somada a pontuação geral do aluno, entretanto, cada atividade pode gerar uma pontuação diferente como é explicado na seção 4.4.4.2. Se o professor cancelar uma fase os pontos obtidos pelos alunos serão invalidados.

4.4.4.2 Pontos

Os pontos são formas quantitativas de recompensar o jogador pelas ações realizadas no ambiente gamificado. Cada ação de sucesso realizada pelo aluno resulta no aumento de pontos, da mesma forma que ações negativas resultam na perda dos pontos. As ações de sucesso se caracterizam como o acerto das tarefas no período proposto pelo professor. Já as ações negativas, são definidas como o não cumprimento das tarefas no período proposto pelo professor.

Cada atividade apresenta um tempo limite para ser realizada, que é definido pelo professor. A realização da tarefa de forma correta no período determinado proporciona ao aluno a quantidade máxima de pontos de uma tarefa. Cada fase tem uma quantidade de pontos definido como Base de Pontos (BP). O valor do BP de cada fase é definido pelo professor e tem seu valor armazenado no banco de regras. Deste modo, o professor tem a liberdade de atribuir uma pontuação maior em fases que sejam mais complexas de serem

resolvidas.

A quantidade de pontos que um aluno pode obter depende da quantidade de tentativas (Tabela 4), bem como, a responsabilidade do aluno em realizar as atividades no período demarcado pelo professor. No cenário em que o aluno não realiza nenhuma tentativa de resolver a tarefa, cem por cento (100%) do valor da BP é descontado, resultando em zero pontos para o aluno de forma instantânea.

Tabela 4 – Relação da quantidade de tentativas ao resolver desafios com a porcentagem de pontos obtidos da Base de Pontos

Quantidade de tentativas	Porcentagem de pontos da Base de Pontos
00	Nenhuma pontuação
01	Cem por cento (100%)
02 a 03	Noventa por cento (90%)
04 a 05	Oitenta por cento (80%)
06 ou mais	Setenta por cento (60%)
01 ou mais após fim do tempo de realização da tarefa	Quarenta por cento (30%)

Nos casos de resolução das atividades no tempo definido, o aluno pode receber as seguintes pontuações: em caso de acerto na primeira tentativa, cem por cento (100%) da BP; se ocorrer duas ou três tentativas, noventa por cento da BP (90%); entre quatro e cinco, oitenta por cento da BP (80%); e em seis ou mais, setenta por cento da BP (60%). Já em caso de realização e acerto da tarefa após o tempo definido, permite ao aluno obter quarenta por cento (40%) do valor da BP independentemente da quantidade de tentativas.

Em uma fase a quantidade máxima de pontos que um aluno pode obter é definida pela quantidade de tarefas vezes o valor da BP. Por exemplo, em uma fase com BP definido em dez pontos e tenha dez tarefas definidas pelo professor, permite ao aluno ter entre zero e cem pontos (Base de Pontos * quantidade de tarefas).

4.4.4.3 Badges

O MAAGIP usa de distintivos como uma forma dos alunos exibirem *status* em casos de conquistas específicas predefinidas pelo modelo gamificado. Os badges do MAAGIP são divididos em três categorias, que são: (1) de bronze, para conquistas simples de serem atingidas; (2) de prata, para conquistas com dificuldade médias; (3) e de ouro, com alta dificuldade de se realizar.

Para conquistar os *badges* o aluno precisa cumprir um requisito, ou seja, realizar uma ação predefinida do modelo gamificado. Os requisitos dos distintivos envolvem ações como: finalizar atividades ou fases antes dos colegas; ser o aluno na primeira posição do *ranking* de uma fase; e colecionar distintivos. O *badge* de bronze, é adquirido sem requerer muito esforço do aluno, simplesmente para apresentar ao jogador a possibilidade de recompensa. Já os *badges* com dificuldade média (prata) e alta (ouro), podem atuar como objetivos que os alunos buscarão atingir, o que demandará a realização de um requisito para cada distintivo. A seguir são listados todas as categorias de *badges* e o requisito que o aluno precisa realizar para conquistá-los.

- Badges de bronze:

Figura 12 – Badges de bronze



Badges de acertos: obtido quando o aluno resolve corretamente uma questão em uma fase pela primeira vez.



Badge de finalização: recebido quando o aluno finaliza uma fase pela primeira vez no jogo.



Badge de posição: estar entre o quarto e o decimo aluno a concluir uma fase.



Badge de ranking: obtido quando o aluno está pela primeira vez entre as dez primeiras posições do ranking.

- Badges de prata:

Figura 13 – Badges de prata



Badges de acerto: obtido quando o aluno consegue resolver na primeira tentativa 30% das atividades de uma fase.



Badge de posição: ser o segundo ou terceiro aluno a concluir uma fase.



Badges de coleção: obtido quando o aluno conquista três badges de bronze.



Badge de ranking: estar pela primeira vez nas cinco primeiras posições do ranking.

- Badges de ouro:

Figura 14 – Badges de ouro



Badges de acerto: obtido quando o aluno consegue resolver na primeira tentativa 50% das atividades de uma fase.



Badge de posição: ser o primeiro aluno a concluir uma fase.



Badges de coleção: obtido quando o aluno conquista três ou mais badges de bronze.



Badge de ranking: conquistado quando o aluno atinge a primeira posição no ranking pela primeira vez.

No MAAGIP os badges são exibidos no perfil do aluno, que será detalhado no tópico 4.4.4.5 a seguir. Os *badges* provém aos alunos a sensação de *status*, que é uma forma de atribuir uma recompensa significativa aos participantes (MEDINA et al., 2014). Deste modo, alunos que apresentam comportamento do tipo conquistador podem se sentir motivados a realizar os requisitos para conquistar os distintivos, o pode fazer com que os alunos se mantenham focados em realizar as atividades de IP.

4.4.4.4 Ranking (Tabela de classificação)

O *ranking*(Classificação) é uma forma de classificar os alunos com base na pontuação obtida ao realizar as atividades de uma fase. A classificação é uma forma de ordenar os alunos conforme o seu desempenho baseado nos pontos obtidos. Assim os alunos são listados em ordem crescente, ou seja, os jogadores com melhores pontuações assumem as

posições altas no *ranking*. A classificação é uma forma de representar o desempenho dos alunos ao longo do período total de uso do sistema gamificado.

Cada fase concluída pelo aluno gera pontos que são somados a sua pontuação geral. Esta pontuação, é adotada para classificar os alunos em uma lista destacando as maiores pontuações de uma turma. O *ranking* é exibido na interface gamificada, o que permite que os participantes da ferramenta gamifica possam comparar suas posições com os concorrentes. Deste modo, os alunos que apresentam características de jogador predador podem se sentir motivados em realizar as atividades buscando melhores posições na classificação.

4.4.4.5 Avatar (Perfil do aluno)

O perfil do aluno ou *avatar*, é a representação dos participantes como jogadores na ferramenta gamificada. No perfil do aluno é listado todas as principais informações do jogador, como: *badges* conquistados ao longo das atividades; a pontuação geral do aluno que representa a soma de todos os pontos obtidos com todas as atividades; os colegas do jogador que fazem parte da mesma turma; a posição nos *rankings* que o aluno se encontra; e por fim as fases e seus respectivos *status*.

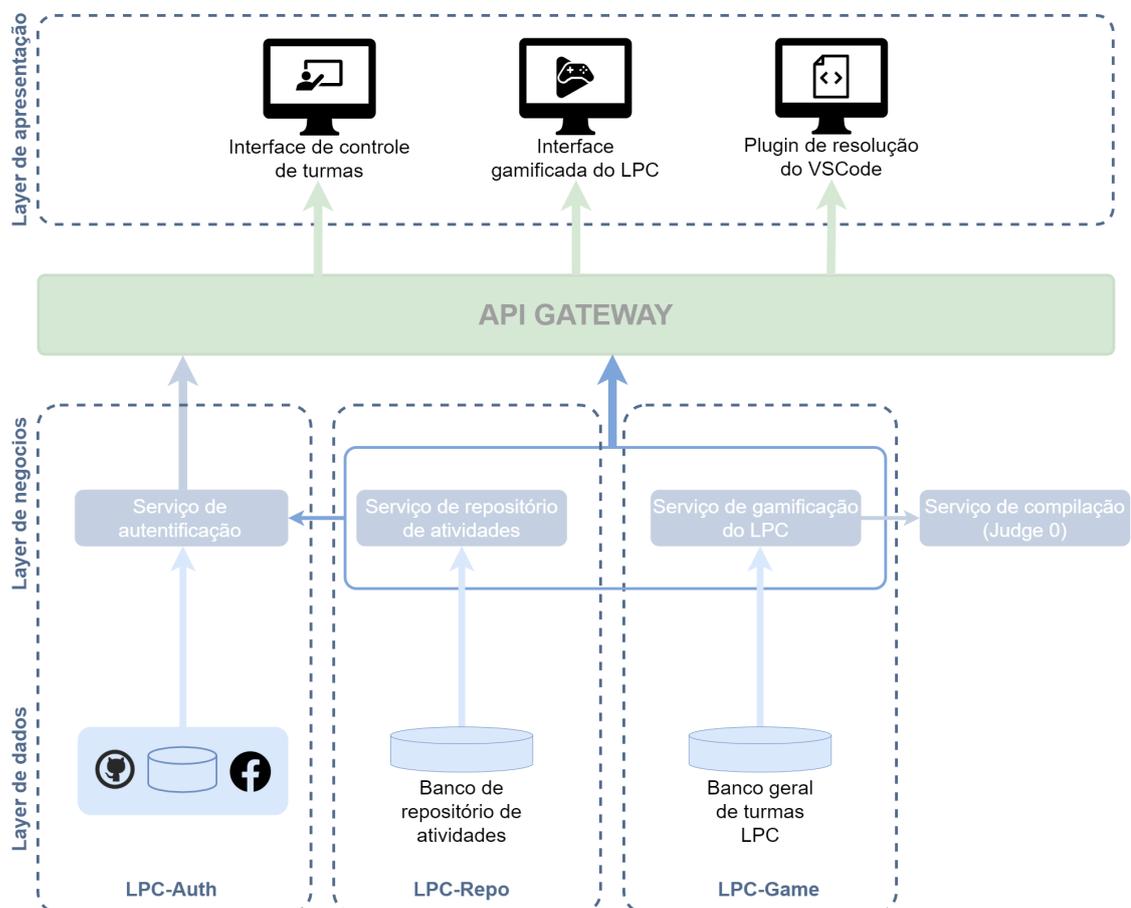
No perfil do aluno uma representação das fases é exibida destacando a quantidade de fases existentes, quantas foram realizadas e quantas faltas para concluir o jogo. Além disso, também é destacado a quantidade de pontos obtidos, além dos *badges* conquistados pelo jogador.

5 FERRAMENTA LPC (LEARN, PLAY AND CODE)

LPC (Learn, Play and Code) (Figura 15), é uma ferramenta criada e desenvolvido nesse trabalho que usa do modelo gamificado MAAGIP (Modelo de Aplicação de Atividades Gamificadas de Introdução à Programação) na gamificação de atividades aplicadas na disciplina de Introdução à programação (IP). A ferramenta LPC é baseada na arquitetura de microsserviços, ou seja, o sistema é dividido em partes. Assim, cada serviço tem sua estrutura lógica e de banco de dados individual.

O LPC é composto por três serviços principais, sendo eles: (1) Autenticação, responsável por validar o acesso dos usuários à ferramenta; (2) Repositório de atividades, que é um banco de desafios de programação propostos e consumidos por professores; (3) Sistema gamificado, que distribui atividades para as turmas de IP e gamifica os resultados das resoluções dos desafios. Também faz parte da ferramenta o Judge0, que é uma ferramenta auxiliar que compila, executa e retorna os resultados dos códigos.

Figura 15 – Arquitetura em três camadas do LPC



A adoção de uma arquitetura baseada em microsserviços pode viabilizar a escala-

bilidade e a flexibilidade do sistema. A divisão em serviços individuais permite o uso de tecnologias de programação distintas para cada microsserviço. Além disso, uma equipe pode atuar nas funcionalidades de um serviço sem causar conflitos no código-fonte de outras partes da ferramenta (REDHAT, 2018).

Para propósito de simplificação, cada microsserviço do LPC será classificado em três camadas (*Layers*) distintas. São elas: *Layer* de apresentação, negócios e dados. A seguir, são definidas as referidas camadas:

- *Layer* de apresentação: se comunica com a layer de negócios para obter dados e apresentá-los aos usuários. Entretanto, não pode interagir diretamente com a *layer* de dados;
- *Layer* de negócios: ou camada lógica da aplicação, é onde se concentram as regras de funcionamento da ferramenta. A camada de negócios é onde a maior parte do processamento ocorre e também é a única *layer* com capacidade de se comunicar com a layer de dados;
- *Layer* de dados: é uma camada protegida que armazena e gerencia todos os dados do LPC e só pode ser acessada pela *layer* de negócios. Cada microsserviço tem sua camada de dados individual que se comunica com a camada lógica do próprio microsserviço.

Para se comunicar, os serviços adotam o padrão de Rest API. Contudo, isso pode tornar complexo para a *layer* de apresentação consumir os microsserviços, pois teria que administrar múltiplas requisições API para serviços com *Endpoints* distintos. Para solucionar esse problema, é adotado um API *Gateway*, que é uma camada que roteia as requisições da camada de apresentação e as redireciona ao serviço correto (ANIL, 2021).

```

1   LPCApi lpcApi = new LPCApi();
2
3   lpcApi.init(
4       appKey: "858049452536",
5       url: "https://lpc.com/api/"
6   );

```

Listing 5.1 – Exemplo de iniciação da lib LPCApi

A lib(biblioteca) LPC-API (*Listing 5.1*) foi desenvolvida para acessar os serviços do LPC, isto é, os microsserviços de autenticação, repositório e gamificação. LPC-API é uma biblioteca desenvolvida na linguagem de programação em Dart, que permite a comunicação com o API *Gateway* do LPC. Graças a LPC-API, a layer de apresentação pode realizar requisições para o LPC sem se preocupar em administrar a comunicação entre os microsserviços. Como exibido na *Listing 5.1*, para instanciar a LPC-API é preciso de

dois parâmetros: (1) a App-key, que é a chave de identificação da aplicação; (2) e a URL, que é o endereço de acesso do *Api Gateway*.

O único microsserviço que estabelece comunicação com todos os microsserviços da layer de negócios é o Serviço de Autenticação, já que este valida o acesso aos demais microsserviços. Desse modo, toda consulta de dados depende da aprovação do LPC-Auth, que é definido na seção a seguir

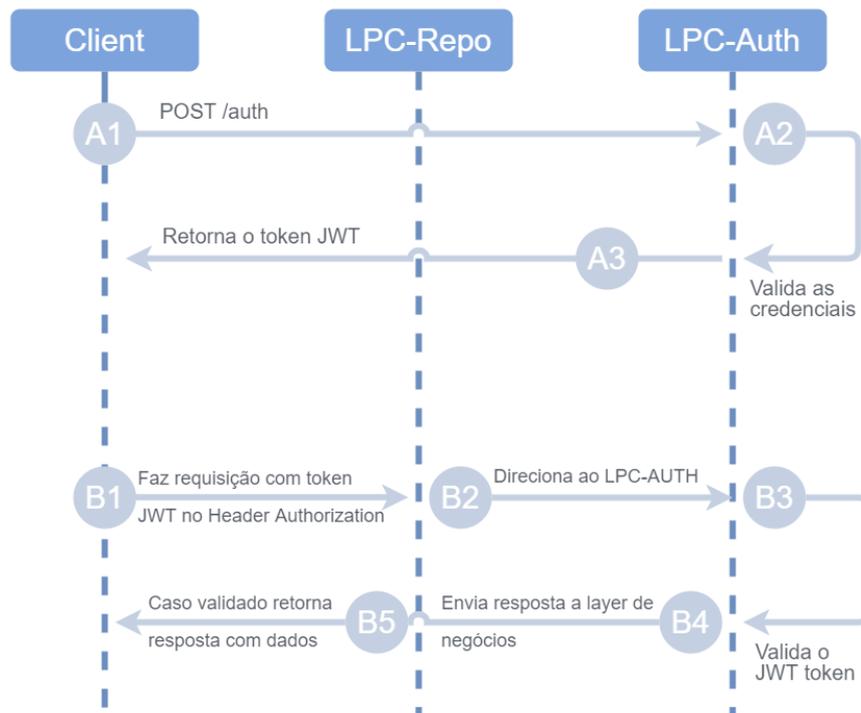
5.1 Microsserviço de autenticação(LPC-AUTH)

O microsserviço de autenticação (LPC-Auth) é responsável pela identificação e autorização de acesso dos usuários à ferramenta LPC. Para isso, o LPC-Auth usa do padrão de *JWT Token* para autenticar o ingresso dos usuários (*Client*), que podem ser professores ou alunos. Além disso, o microsserviço de autenticação é usado para validar o acesso aos dados da ferramenta. Assim, o cliente requisita acesso a layer lógica que usa o LPC-Auth para validar e retornar as informações.

O LPC-Auth é um microsserviço individual com responsabilidade voltada apenas para autenticação e validação de usuários. Ou seja, o LPC-Auth não assume nenhuma outra função na ferramenta LPC que não envolva os usuários e suas credenciais de acesso. Para isso, a camada de negócios recebe uma requisição e consulta a camada de dados para buscar informações dos usuários.

O LPC-Auth pode usar tanto o banco de dados local para salvar as informações do cliente, quanto serviços de autenticação de terceiros, como Github, Facebook e Google. Para isso, a camada lógica do LPC-Auth direciona a requisição para as ferramentas de autenticação externas e aguarda a resposta. Por fim, o microsserviço de autenticação envia uma resposta a quem realizou a requisição.

Figura 16 – Ciclo de autenticação



Na figura 16 é exibido os fluxos principais que a ferramenta realiza para autenticar e validar o acesso às informações da ferramenta. A seguir é descrito o primeiro fluxo (Fluxo A), destacando como acontece as etapas da autenticação de um usuário:

- (A1) O usuário realiza uma requisição de *POST* do padrão REST-API passando as credenciais para de login e senha em formato JSON (*Listing 5.2*), que são direcionadas ao LPC-Auth;

```

1  {
2      "username": "user",
3      "password": "senha123"
4  }
  
```

Listing 5.2 – Exemplo de JSON com as credenciais de autenticação

- (A2) Após isso, a *layer* lógica do LPC-Auth acessa a *layer* de dados e valida as credenciais recebidas. Desse modo, gera um *token* caso as credenciais sejam válidas;
- (A3) Por fim, uma resposta com o token de acesso (*JWT token*), tipo do *token*, data de expiração, e *Refresh Token* (*Token* de revalidação) são retornados ao cliente caso a validação seja aprovada (*Listing 5.3*). Se a autenticação for inválida, uma resposta negativa é retornada.

```

1 {
2   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXV... ",
3   "type": "Baerer",
4   "expireAt": "Sunday, 10 January 2021 13:51:26",
5   "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXV..."
6 }

```

Listing 5.3 – Exemplo resposta do POST de autentificação

O Fluxo B (Figura 16) representa as etapas para obter acesso aos dados. Esses dados são todas as informações às quais o usuário pode ter acesso na ferramenta, como desafios, turmas, pontuações, dentre outros. Diferente do fluxo de autentificação que o cliente efetua, uma requisição direta ao LPC-Auth, no fluxo de acesso aos dados, a requisição do usuário é direcionada à camada de negócios dos demais microsserviços. Abaixo são explicadas as etapas do segundo fluxo:

- (B1) O usuário usa o *JWT Token* no Cabeçalho de autorização (*Header Authorization*) para realizar uma requisição de dados, que é direcionada a *layer* de negócios, que pode ser do sistema gamificado ou do sistema de repositórios;
- (B2) A *layer* lógica recebe uma requisição requerendo acesso de dados. Desse modo, a LPC-Auth é chamada para analisar se o acesso do usuário é válido;
- (B3) O LPC-Auth retorna uma resposta positiva caso o *JWT Token* seja válido;
- (B4) A camada de negócios recebe a resposta do LPC-Auth e, caso seja positiva, retorna com os dados que o cliente requisitou;
- (B5) Caso todo processo ocorra sem erros, e o *JWT Token* seja válido, o usuário recebe os dados que requisitou em formato JSON.

Além dos fluxos principais, outras funções básicas do LPC-Auth incluem o cadastro de alunos e professores. Também é possível atualizar senha ou requisitar a recuperação da credencial de acesso. O LPC-Auth mantém as informações cadastrais que são consumidas pelos microsserviços gamificados e o repositório. Essas informações do usuário são registros de cadastro e registros de autentificação.

5.2 Microsserviço de repositório de atividades

O microsserviço de repositório de atividades (LPC-Repo) é o serviço responsável por armazenar e catalogar todos os desafios de programação do LPC. O LPC-Repo age

como um repertório de desafios de programação, o que permite que os professores busquem por atividades para aplicar em sala de aula. Também é possível que os educadores proponham novos desafios e os disponibilizem no repositório. A seguir são listadas todas as funções do LPC-Repo:

- Propor novos desafios e disponibilizá-los no LPC-Repo. Ao desenvolver um novo desafio, o professor pode anexar a este uma ou mais palavras-chave que descrevam o desafio. Para evitar inconsistências, como no caso de cada professor conceber diferentes termos que representem o mesmo conteúdo, o LPC-Repo dispõe de um conjunto de palavras-chave que podem ser anexados aos desafios;
- Buscar desafios por palavras-chave que representem os conteúdos ensinados em sala de aula. Exemplo: variáveis, laço de repetição, classes e outros;
- Editar um desafio já existente, o que permite que um professor altere uma questão já existente no repositório para aplicar em sua turma;
- Criar conjuntos de desafios (Fase) e disponibilizar no repositório. Dessa forma, um professor pode desenvolver uma nova fase com um grupo de atividades, e dispor no repositório. Assim, outros professores podem realizar buscas por Fases completas para aplicar em suas aulas;
- Buscar grupos de desafios já existentes no repositório, além da capacidade de editar uma fase já existente, caso o professor precise adicionar ou remover desafios. A busca de Fases usa palavras-chave, do mesmo modo que ocorre na busca de desafios.

Para acessar as funcionalidades do LPC-Repo é utilizado a biblioteca LPC-API, que dispõe de uma subclasse específica para se comunicar com o LPC-Repo. Como exibido na *Listing 5.4*, a LPC-API retorna uma instância da classe `LPCRepository`, que pode realizar requisições para serviço de repositório.

```
1 LPCRepository lpcRepository = lpcApi.repositoryInstance;  
2  
3 Challenge challenge = lpcRepository.filterByID('123');
```

Listing 5.4 – Filtrando desafio com classe LPCRepository

Como exibido na linha três (*Listing 5.4*) é possível realizar uma busca de desafios através do ID (Identificador). Caso o ID seja válido, o `LPCRepository` retornará uma instância de uma classe do tipo `Challenge` (Desafio). Na *Listing 5.5* é possível observar os dados que compõem a classe do tipo `Challenge` no formato JSON. A classe de desafio contém informações, como: título e descrição do desafio; palavras-chave; data de criação e última edição do desafio; entradas fornecidas; e saídas esperadas.

```
1 {
2   "id": "1565",
3   "created-date": "2022-01-13 18:21:43",
4   "edited-date": "2022-01-13 18:21:43",
5   "title": "Média aritmética",
6   "description": "Dado três valores de entrada, crie um
7     algoritmo que retorne a média aritmética",
8   "key-words": [
9     "Operações matemáticas"
10  ],
11  "codeEntries": [
12    {
13      "variable": "number01",
14      "value": "10",
15      "type": "double"
16    },
17    {
18      "variable": "number02",
19      "value": "7",
20      "type": "double"
21    },
22    {
23      "variable": "number03",
24      "value": "4",
25      "type": "double"
26    }
27  ],
28  "expectedOutputs": [
29    {
30      "position": 0,
31      "value": 7,
32      "type": "double"
33    }
34  ]
35 }
```

Listing 5.5 – Arquivo JSON de um desafio

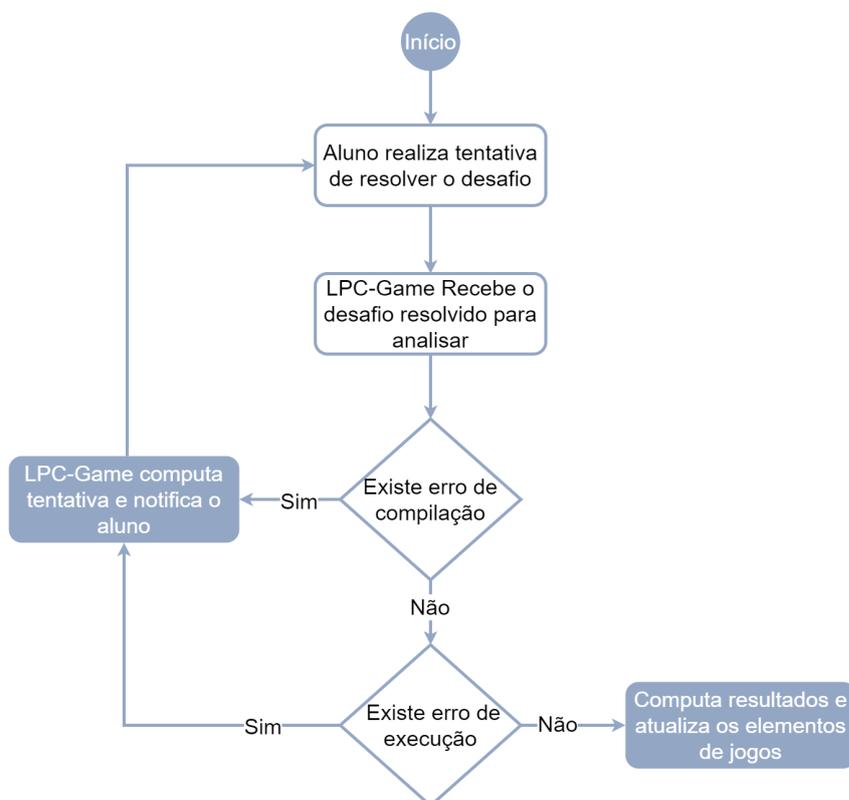
A estrutura dos desafios contém entradas fornecidas e saídas esperadas (*Listing*

5.5). As entradas fornecidas são valores que os professores agregam aos desafios que serão usados para validar os algoritmos desenvolvidos pelos alunos. Na linha dez são fornecidas três entradas, que são os valores dez(10), sete(07) e quatro(04). O desafio pede que os alunos criem um algoritmo que realize a média aritmética dos três valores de entrada e retorne o resultado. Com o desafio resolvido, o resultado gerado pelo algoritmo do aluno é então comparado com as saídas esperadas, que contém como valor a resposta correta para o desafio. Desse modo, o microserviço de sistema gamificado analisa os resultados e computa pontos positivos para o aluno em caso de acerto.

5.3 Microserviço de sistema gamificado

LPC-Game é um microserviço do LPC criado por esse trabalho, que é responsável por gamificar os resultados dos desafios de Introdução à Programação. Essa gamificação consiste em: transformar um conjunto de atividades em fases; converter resoluções corretas de desafios em pontuações; conquistas realizadas pelos alunos se convertem em *badges* (medalhas); alunos com boas pontuações assumem melhores posições no *ranking*; e as ações dos alunos na ferramenta geram *feedbacks*. Além de gamificar as atividades de IP, o LPC-Game armazena em seu banco de dados as informações sobre os alunos que compõem as turmas.

Figura 17 – Algoritmo de gamificação do LPC-GAME



A Figura 17 exhibe o algoritmo que o LPC-Game adota para gamificar os resultados

dos alunos. O processo de gamificação inicia na tentativa dos alunos em resolver uma atividade. Para isso, os alunos vão usar o LPC-Code, que é uma extensão da ferramenta VSCode. Ao finalizar um desafio, o aluno submeterá sua resposta que será enviada ao LPC-Game para ser analisada.

Quando o LPC-Game recebe uma resposta de algum desafio, inicia-se o processo de análise. A primeira parte da análise consiste em usar a ferramenta Judge0 para compilar o algoritmo proposto pelo aluno. Caso ocorra algum erro na compilação, o LPC-Game retorna para o aluno um feedback negativo. Um erro de compilação pode ocorrer por inconsistência sintática, ou seja, um erro na gramática do código. Também é possível que ocorra uma falha semântica, que pode ser um erro na lógica do código.

Caso não exista um erro na compilação, o LPC-Game avança para a próxima etapa, que é executar o algoritmo desenvolvido pelo aluno. Para isso, o resultado da compilação, ou seja, um arquivo executável, recebe as entradas fornecidas pelo desafio e executa o algoritmo. O resultado final da resposta do aluno é então comparado com a saída esperada do desafio. Caso a resposta final não seja igual à saída esperada, o LPC-game retorna um feedback negativo para o aluno.

No caso de não haver erros na execução da resposta do aluno, o LPC-Game inicia o processo de computar os resultados e atualizar os elementos gamificados. Desse modo, o LPC-Game pontua o aluno com base no tempo e na quantidade de tentativas que este precisou para resolver o desafio. Em sequência, a posição do aluno no ranking é atualizada, caso sua pontuação supere a de seus concorrentes. Com o ranking atualizado, o LPC-Game premia os alunos com badges, caso alguma conquista especificada no capítulo 4.4.4.3 tenha sido realizada. Por fim o LPC-Game envia ao aluno um feedback positivo especificando todas as conquistas realizadas.

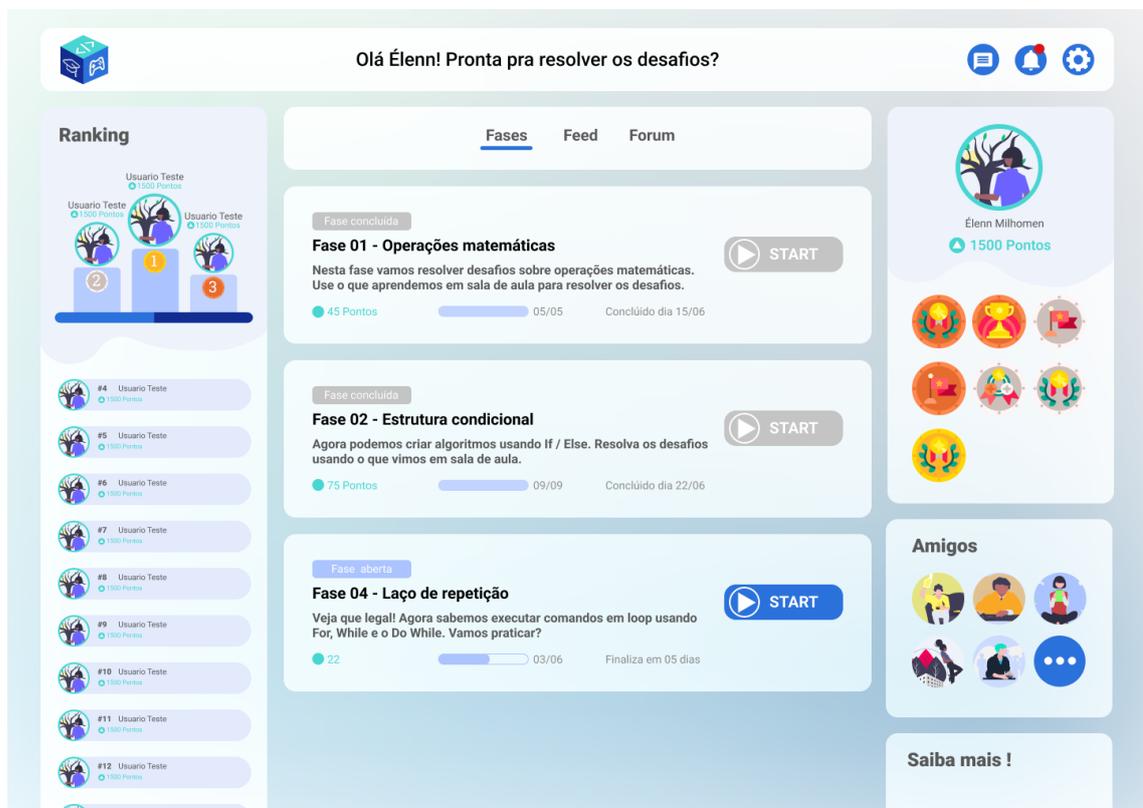
5.4 Layer de apresentação

Na camada da layer de apresentação (LPC-Admin e LPC-Page) estão contidas todas as ferramentas visuais do LPC, ou seja, as páginas *web* que permitem o acesso de alunos e professores. A layer de apresentação é composta por quatro ferramentas visuais: (1) Interface gamificada do LPC, que apresenta os dados gamificados aos alunos; (2) *Extension* de resolução do VSCode, que é uma aplicação que permite ao aluno resolver desafios. Para isso, é integrado ao Visual Studio Code uma extensão de funcionalidades do LPC; (3) Interface de controle de turmas, que permite ao professor moderar as turmas e acessar o repositório de atividades, podendo, assim, propor ou pesquisar por novos desafios.

5.4.1 Interface gamificada da LPC

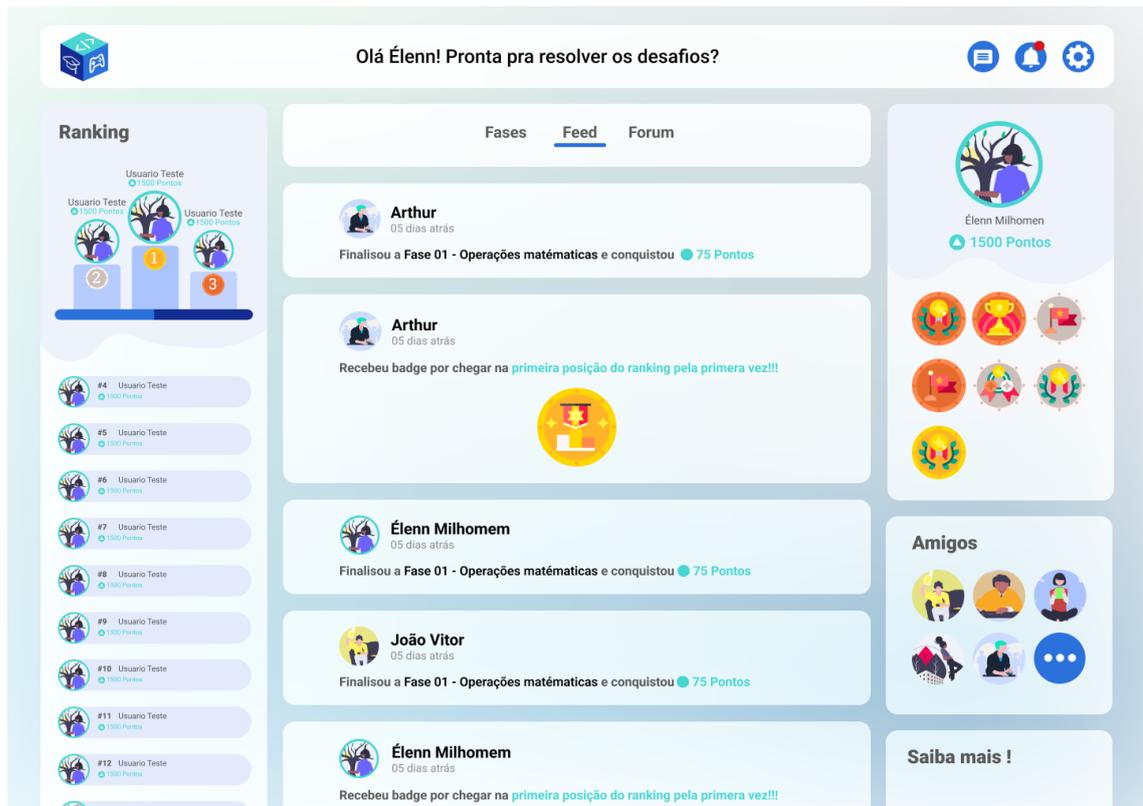
A interface gamificada da LPC (LPC-Page)(Figura 18) é uma página web que os alunos de introdução à programação poderão utilizar para realizar os desafios de programação. O LPC-Page exibe em sua página principal os elementos definidos do MAAGIP. São eles, pontuação, *ranking*, *ranking*, fases, barra de progresso, avatar (Perfil do aluno) e *badges* (Medalhas). Outro elemento de jogo aplicado é o *feedback*, ou seja, uma resposta positiva ou negativa das ações dos alunos.

Figura 18 – Página de fases do protótipo da ferramenta LPC



Na LPC-Page está contida uma seção de *feed* (Figura 19), que lista as ações de todos os alunos de uma turma. Essas ações podem ser conclusão de fases, conquistas de *badges* e mudanças de posição no *ranking*. O *Feed* permite que os alunos acompanhem o que está acontecendo no mundo do jogo (Turma de IP), além de ter noção das jogadas de outros jogadores (outros alunos).

Figura 19 – Página de feed do protótipo da ferramenta LPC

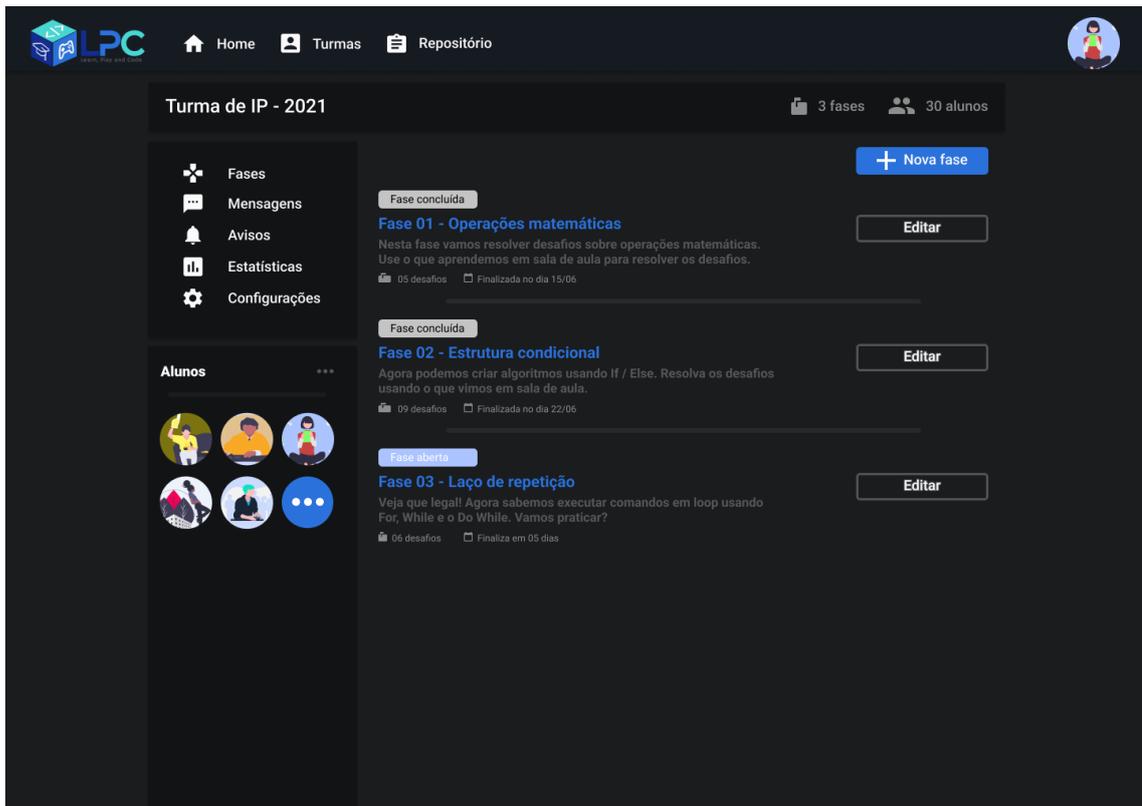


A LPC-Page tem também como ferramenta extra um Fórum, que é uma seção que permite os alunos se comuniquem entre si e possa interagir no ambiente do LPC. Outra função do Fórum é exibir avisos que os professores postarem. Também é possível que os alunos postem dúvidas, que podem ser respondidas por outros participantes ou até mesmo pelos professores.

5.4.2 Interface de controle de turmas

Interface de controle de turmas (LPC-Admin)(Figura 20) é uma página web usada para administrar as turmas de IP na ferramenta LPC. Através do LPC-Admin os professores podem criar e gerir os desafios aplicados em sala de aula. Outra função da página de administração é acompanhar a evolução dos alunos ao resolverem as atividades. Para isso, existe uma seção de estatísticas na interface do professor.

Figura 20 – Página de controle de turmas do protótipo da ferramenta LPC



No momento que uma nova turma é criada através do LPC-Admin, um convite em formato de URL (*web link*) é gerado. Por meio desse *link*, o professor pode convidar alunos para ingressarem em uma nova turma. Com a nova turma criada, o professor pode: propor fases; editar fases existentes; acompanhar a evolução da turma; responder dúvidas dos alunos através da seção de mensagens; e usar da função de avisos para postar mensagens importantes para os alunos.

6 CONCLUSÃO

Diversos trabalhos de outros autores mostram que a gamificação pode apresentar resultados positivos no contexto educacional. Isso ocorre devido aos elementos de jogos, que se bem aplicados podem aumentar a motivação de alunos.

Deste modo, o presente trabalho teve como objetivo propor o protótipo de uma ferramenta gamificada voltada para a aplicação de atividade da disciplina de Introdução de Programação (IP). Para isso, o modelo gamificado MAAGIP foi desenvolvido para estruturar como os elementos de jogos atuarão no contexto de aplicação de atividades de IP. Com o modelo gamificado definido, a ferramenta LPC (Learn, Play and Code) foi concebida com uma arquitetura baseada em microsserviços.

A prática de programação através de exercícios é um fator que pode contribuir no aprendizado de programação. Deste modo, o LPC foi pensado como uma possível solução para manter os alunos motivados ao realizarem as atividades de programação. Para isso, elementos como, pontuação, *ranking*, *badges*, *feedback* e fases foram adotados. Isso com o intuito de trazer diversão para as tarefas realizadas ao longo da disciplina de Introdução a Programação. E talvez como resultado, ocorra um melhor aprendizado dos conceitos de IP.

Uma das últimas etapas realizadas ao propor uma ferramenta gamificada, é a validação e evolução da ferramenta, entretanto, essa última etapa não faz parte do presente trabalho. Deste modo, fica como trabalho futuro a validação da ferramenta LPC, que envolve analisar se os elementos de jogos aplicados obtiveram êxito em motivar os alunos na prática de programação. E por fim, a evolução da ferramenta que deve considerar os pontos negativos do LPC através da análise realizada. Deste modo, melhorias devem ser realizadas para evoluir a capacidade do LPC em motivar os alunos.

REFERÊNCIAS

- ALVES, F. **Gamification. Como Criar Experiências de Aprendizagem Engajadoras. Um Guia Completo. Do Conceito à Prática.** 2^a edição. ed. São Paulo: DVS Editora, 2015. ISBN 978-85-8289-088-2.
- ANIL, N. **Padrão de gateway de API versus comunicação direta de cliente com microsserviço.** 2021. <<https://docs.microsoft.com/pt-br/dotnet/architecture/microservices/architect-microservice-container-applications/direct-client-to-microservice-communication-versus-the-api-gateway-pattern>>. Acessado em 02/10/2021.
- BARTLE, R. **Designing Virtual Worlds.** [S.l.]: New Riders Games, 2003. ISBN 0131018167.
- BOSSE, Y.; GEROSA, M. A. Reprovações e trancamentos nas disciplinas de introdução à programação da universidade de são paulo: Um estudo preliminar. In: . [S.l.: s.n.], 2015. p. 426–435.
- BRITO, A. L. de S. **Level up: uma proposta de processo gamificado para a educação.** Tese (Doutorado) — UFRN - Programa de Pós-Graduação em Engenharia de Software – PPGSW, São Carlos, 06 2017.
- BURKE, B. **Gamificar: Como a gamificação motiva as pessoas a fazerem coisas extraordinárias.** DVS Editora, 2015. ISBN 9788582891148. Disponível em: <<https://books.google.com.br/books?id=33OwDQAAQBAJ>>.
- CAHYANI, A. Gamification approach to enhance students engagement in studying language course. **MATEC Web of Conferences**, v. 58, p. 03006, 01 2016.
- DETERDING, S. et al. From game design elements to gamefulness: Defining "gamification". In: **Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments.** New York, NY, USA: ACM, 2011. (MindTrek '11), p. 9–15. ISBN 978-1-4503-0816-8. Disponível em: <<http://doi.acm.org/10.1145/2181037.2181040>>.
- DICHEV, C.; DICHEVA, D. Gamifying education: what is known, what is believed and what remains uncertain: a critical review. **International Journal of Educational Technology in Higher Education**, v. 14, 02 2017.
- DICHEVA, D. et al. Gamification in education: A systematic mapping study. **Educational Technology Society**, v. 18, p. 75–88, 07 2015.
- DUOLINGO. 2021. <<https://pt.duolingo.com/>>. Acessado em 18/05/2021.
- FARDO, M. L. The multiplayer classroom: Designing coursework as a game. In: **Revista do Programa de Pós- Graduação em Educação da UFPI.** [S.l.: s.n.], 2014. p. 269–273.

FOGG, B. A behavior model for persuasive design. In: **Proceedings of the 4th International Conference on Persuasive Technology**. New York, NY, USA: ACM, 2009. (Persuasive '09), p. 40:1–40:7. ISBN 978-1-60558-376-1. Disponível em: <<http://doi.acm.org/10.1145/1541948.1541999>>.

FOGG, B. A behavior model for persuasive design. In: **Proceedings of the 4th International Conference on Persuasive Technology**. New York, NY, USA: Association for Computing Machinery, 2009. (Persuasive '09). ISBN 9781605583761.

FREDRICKS, J. A.; FILSECKER, M.; LAWSON, M. A. Student engagement, context, and adjustment: Addressing definitional, measurement, and methodological issues. **Learning and Instruction**, v. 43, p. 1 – 4, 2016. ISSN 0959-4752. Special Issue: Student engagement and learning: theoretical and methodological advances. Disponível em: <<http://www.sciencedirect.com/science/articel/pii/S0959475216300159>>.

GOSHEVSKI, D.; VELJANOSKA, J.; HATZIAPOSTOLOU, T. A review of gamification platforms for higher education. In: **Proceedings of the 8th Balkan Conference in Informatics**. New York, NY, USA: ACM, 2017. (BCI '17), p. 28:1–28:6. ISBN 978-1-4503-5285-7. Disponível em: <<http://doi.acm.org/10.1145/3136273.3136299>>.

HESS, T.; GUNTER, G. Serious game-based and nongame-based online courses: Learning experiences and outcomes. **British Journal of Educational Technology**, v. 44, 05 2013.

KAPP, K. M. **The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education**. 1st. ed. [S.l.]: Pfeiffer & Company, 2012. ISBN 1118096347, 9781118096345.

KHAN Academy. 2021. <<https://www.khanacademy.org/>>. Acessado em 20/05/2021.

LAZZARO, N. Why we play games: Four keys to more emotion without story. **Game Dev Conf**, 01 2004.

LOMBRISER, P.; VALK, R. van der. Improving the quality of the software development lifecycle with gamification. In: . [S.l.: s.n.], 2014.

MARINS, D. R. **Um processo de gamificação baseado na teoria da autodeterminação**. Tese (Doutorado) — USP - Instituto de Ciências Matemáticas e de Computação, Universidade Federal do Rio de Janeiro, 12 2013.

MEDINA, B. et al. **Como Reinventar Empresas a Partir de Jogos**. [S.l.]: Mjv Press, 2014. ISBN 8565424081.

PEDRO, L. Z. **Uso de gamificação em ambientes virtuais de aprendizagem para reduzir o problema da externalização de comportamentos indesejáveis**. Tese (Doutorado) — USP - Instituto de Ciências Matemáticas e de Computação, São Carlos, 3 2016.

PINTO, F. de S.; SILVA, P. C. Gamification applied for software engineering teaching-learning process. In: **Proceedings of the 31st Brazilian Symposium on Software Engineering**. New York, NY, USA: ACM, 2017. (SBES'17), p. 299–307. ISBN 978-1-4503-5326-7. Disponível em: <<http://doi.acm.org/10.1145/3131151.3131186>>.

REDHAT. **O que é arquitetura de microsserviços?** 2018. <<https://www.redhat.com/pt-br/topics/microservices/what-are-microservices>>. Acessado em 02/10/2021.

RIBEIRO, R. B. S. **UTILIZANDO GAMIFICAÇÃO EM UM SISTEMA DE JUIZ ONLINE PARA ENGAJAR ALUNOS DE GRADUAÇÃO EM DISCIPLINAS INICIAIS DE PROGRAMAÇÃO.** Dissertação (Mestrado), 2018.

ROCHA, P. et al. Ensino e aprendizagem de programação: Análise da aplicação de proposta metodológica baseada no sistema personalizado de ensino. **RENOTE**, v. 8, 03 2010.

RYAN, R. M.; DECI, E. L. Overview of self-determination theory: An organismic-dialectical perspective. In: . [S.l.: s.n.], 2002.

SCHELL, J. **The Art of Game Design: A Book of Lenses.** [S.l.]: Elsevier Inc., 2008.

SCHOEFFEL, P. et al. Mapping factors that impact on motivation of undergraduate students in software engineering: A case study. In: . [S.l.: s.n.], 2018. p. 963.

SHELDON, L. **The Multiplayer Classroom: Designing Coursework as a Game.** [S.l.]: Cengage Learning, 2011. ISBN 9781435458444.

SILVA, F.; BAX, M. Gamificação na educação online: proposta de modelo para a aprendizagem participativa. **Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação**, v. 22, n. 50, p. 144–160, 2017. ISSN 1518-2924. Disponível em: <<https://periodicos.ufsc.br/index.php/eb/article/view/1518-2924.2017v22n50p144>>.

TED. **Jane McGonigal - Gaming can make a better world.** 2010. Último acesso em 01/08/2019. Disponível em: <https://www.ted.com/talks/jane_mcgonigal_gaming_can_make_a_better_world>.

TOLOMEI, B. A gamificação como estratégia de engajamento e motivação na educação. **EAD EM FOCO**, v. 7, n. 2, 2017. ISSN 2177-8310. Disponível em: <<http://eademfoco.cecierj.edu.br/index.php/Revista/article/view/440>>.

VIANNA, Y. et al. **Gamification, Inc. Como reinventar empresas a partir de jogos.** [S.l.]: MJV Press, 2013.

WANG, W.; LV, J. A case study of using gamification to improve art education in college class. In: **Proceedings of the 10th International Conference on Education Technology and Computers.** New York, NY, USA: ACM, 2018. (ICETC '18), p. 7–11. ISBN 978-1-4503-6517-8. Disponível em: <<http://doi.acm.org/10.1145/3290511.3290537>>.

WEISER, P. et al. A taxonomy of motivational affordances for meaningful gamified and persuasive technologies. In: **Proceedings of EnviroInfo and ICT for Sustainability 2015.** Atlantis Press, 2015/09. p. 271–280. ISBN 978-94-62520-92-9. ISSN 2352-538X. Disponível em: <<https://doi.org/10.2991/ict4s-env-15.2015.31>>.

WERBACH, K.; HUNTER, D. **How game thinking can revolutionize your business.** [S.l.]: Wharton digital press, 2012.