



APRIMORAMENTO SISTEMÁTICO DO PBL NA ENGENHARIA DE SOFTWARE: UM MÉTODO BASEADO EM OBJETIVOS DE APRENDIZAGEM E VISÕES ARQUITETURAIS

DOI: 10.37702/2175-957X.COBENGE.2025.6200

Autores: REGINALDO ARAKAKI,VANESSA NUNES,HERMANO PEIXOTO DE OLIVEIRA JUNIOR,JOSE ROMUALDO DA COSTA FILHO,JULIA STATERI,GERALDO VASCONCELLOS,MONICA ANASTASSIU

Resumo: *This work details a methodology to review the learning modules backlog named LBL based on Project Based Learning (PBL), aligned with curricular rules and directions of MEC (Ministério de Educação) and PPC (Projeto Pedagógico do Curso) of Software Engineering Bachelor, considering industry professional profiles expected in terms of capacity, ability, and knowledge to build and maintain digital software platforms. This research drew on the experiences and disciplines of business, leadership, computation, mathematics, physics, and user experience experts. Scientific architecture references include ISO/IEC 25010, ISO/IEC 12207, and ISO/IEC 10746, which encompass five viewpoints of a digital system: Business Drivers, Functional and Non-Functional Requirements, Engineering Solutions, and Technology Tools. Such dimensions of approach review are compatible with engineering aspects of modern digital platforms that lead with the human body, transportation devices, and connected cities all d*

Palavras-chave: *Keywords: Curriculum Alignment,Bloom's Taxonomy,Architecture Viewpoints,Modular Design,Learning Assessment,ISO/IEC 42010,Soft and Hard Skills,LBL (Learning Backlog),Project Based Learning.*

APRIMORAMENTO SISTEMÁTICO DO PBL NA ENGENHARIA DE SOFTWARE: UM MÉTODO BASEADO EM OBJETIVOS DE APRENDIZAGEM E VISÕES ARQUITETURAIS

1 INTRODUÇÃO

O ensino superior enfrenta desafios crescentes para preparar profissionais capazes de lidar com a complexidade e a rápida evolução do mercado de trabalho, especialmente nas áreas de tecnologia e engenharia. Nesse contexto, o Project-Based Learning (PBL) emerge como uma metodologia promissora, capaz de proporcionar uma formação mais abrangente e alinhada às demandas contemporâneas (Barrows, 1996). Neste cenário de inovação educacional, destaca-se o Instituto de Tecnologia e Liderança (Inteli), uma instituição de ensino superior pioneira no Brasil na adoção integral do modelo PBL.

A estrutura curricular do Inteli é organizada em módulos de aprendizagem chamados LBLs (Learning BackLogs), que são constantemente revisados e ajustados para garantir a relevância e eficácia do ensino. Estes módulos são projetados para integrar conhecimentos técnicos de computação, matemática e física com competências de liderança, negócios e design de experiência do usuário, proporcionando uma formação holística aos estudantes.

Um aspecto fundamental dessa estrutura é a forma como as Diretrizes Curriculares Nacionais (DCNs) são incorporadas e distribuídas ao longo dos LBLs. As disciplinas tradicionais, conforme definidas nas DCNs, são cuidadosamente decompostas em pacotes de assuntos que são então permeados e revisitados pelos diferentes LBLs. O diferencial dessa abordagem está na forma como os assuntos são apresentados: eles reaparecem em diversos momentos do curso, sendo progressivamente aprofundados ou examinados sob diferentes perspectivas a cada iteração. Essa estratégia de revisão e aprofundamento gradual permite que os estudantes construam um entendimento mais sólido e multifacetado dos temas. Desta forma, o Inteli consegue manter total aderência às exigências legais e às disciplinas definidas nas DCNs, enquanto oferece uma experiência de aprendizagem mais fluida e conectada.

Neste contexto específico do Inteli, mas com implicações para a implementação do PBL no ensino superior em geral, emergem questões importantes sobre a implementação e aprimoramento desta metodologia:

- Como realizar a evolução sistemática dos módulos de aprendizagem baseados em projetos (LBLs)?
- Quais aspectos da aprendizagem devem ser utilizados como base para compor os requisitos de revisão?
- Como os pacotes de conceitos e práticas podem ser reconfigurados sem prejuízo das disciplinas essenciais definidas nas Diretrizes Curriculares Nacionais estabelecidas pelo MEC?

1.1 Objetivo do Artigo

Apresentar uma metodologia e um estudo de caso de aplicação voltada para revisar e aprimorar um módulo, como um exemplo prático da apresentação da metodologia combinando a taxonomia de Bloom com as dimensões arquiteturais de sistemas de software distribuídos, baseadas nas normas como a ISO/IEC 42010. Obviamente direcionado para os aspectos de liderança do Inteli em termos socioemocionais necessários ao perfil profissional do Engenheiro de Software.

2 REVISÃO BIBLIOGRÁFICA SOBRE PBL

O Project-Based Learning (PBL) tem sido amplamente estudado e implementado em diversas instituições de ensino superior ao redor do mundo. Esta seção apresenta uma revisão da literatura sobre o uso do PBL no ensino superior, com ênfase em sua aplicação na área de Engenharia de Software.

Alguns trabalhos que estão relacionados com o PBL, evidenciam a oportunidade da proposta de sistematização deste artigo: O desafio da inovação e da melhoria contínua está bem enriquecido nas publicações como as indicadas nas publicações (Felder *et al*, 2004), (Habimorad, 2024), (Markham *et al*, 2008). As experimentações relacionadas à melhoria e eficiência dos projetos PBL na área de engenharia, os resultados obtidos e destaques de melhoria podem ser extraídas das publicações (Hayashi *et al*, 2021), (Hayashi *et al*, 2020), (Cognasca *et al*, 2024) que desafiaram a pandemia para trazer a imersão virtual em experiências práticas de laboratório de circuitos digitais, usando IoT e placas de circuitos de hardware FPGA. Em (Hayashi *et al*, 2023), (Arakaki *et al*, 2021) os *soft skills* e *hard skills* foram destacadas e medidas como aspectos importantes na aprendizagem do engenheiro. As publicações de (Apeanti *et al*, 2021), (Fioravante *et al*, 2018) e (Zhang e Hu, 2024) permitiram um panorama sobre a busca continuada de avaliação de aprendizagem e melhorias relacionados, baseada em medidas de resultados e de eficácia das práticas aplicadas em atividades colaborativas.

3 ADOÇÃO DE PBL NO INTELI E PERSPECTIVAS DE APRENDIZAGEM

Este trabalho propõe uma nova abordagem para aprimorar a implementação do PBL em cursos de engenharia de software, utilizando duas estruturas conceituais complementares. Primeiramente, apresentamos o uso da taxonomia de Bloom como instrumento para projetar e implementar os Learning Backlogs (LBLs), considerando os níveis de aprendizado desejados. Em seguida, introduzimos uma adaptação das dimensões de aprendizado baseadas em visões arquiteturais de sistemas de software distribuídos, derivadas das normas ISO/IEC 42010 e ISO/IEC 10746. A combinação dessas abordagens como um framework para o design, implementação e avaliação de módulos, com uma visão sistêmica da jornada de aprendizado de engenharia de software.

3.1 Taxonomia de Bloom como instrumento de design e evolução de LBLs

A taxonomia de Bloom é utilizada nesta proposta como um instrumento para projetar, implementar e evoluir os Learning Backlogs (LBLs). Esta abordagem permite uma estruturação sistemática dos conteúdos, atividades e avaliações, garantindo coerência e integração dos mecanismos de ensino e aprendizagem ao longo do curso. Os seis níveis da taxonomia de Bloom aplicados são:

- Memorizar: Reconhecer e recordar informações relevantes;
- Compreender: Interpretar e explicar ideias ou conceitos;
- Aplicar: Usar informações em novas situações;
- Analisar: Distinguir entre as diferentes partes e examinar a estrutura;
- Avaliar: Justificar uma decisão ou curso de ação;
- Criar: Produzir trabalho original ou propor soluções alternativas.

Propõe-se o uso destes seis níveis no contexto dos LBLs, de acordo com o conteúdo apresentado na Tabela 1.

Tabela 1 - Proposta de uso da Taxonomia de Bloom.

Memorizar	MMB	Básico	Recorda informações básicas, como fatos simples, definições ou datas importantes.
	MMI	Intermediário	Lembra-se de detalhes específicos e pode recitar informações com alguma precisão.
	MMA	Avançado	Recupera informações detalhadas sem auxílio e demonstra domínio completo dos fatos.
Compreender	COB	Básico	Explica conceitos em suas próprias palavras e consegue resumir informações essenciais.
	COI	Intermediário	Interpreta informações, traduz ideias para diferentes contextos e fornece exemplos adequados.
	COA	Avançado	Analisa implicações dos conceitos e consegue prever resultados baseados em sua compreensão.
Aplicar	APB	Básico	Utiliza conhecimentos adquiridos em situações familiares com orientação.
	API	Intermediário	Aplica conceitos em novos contextos com pouca ou nenhuma assistência.
	APA	Avançado	Resolve problemas complexos de forma autônoma, transferindo conhecimentos para situações inéditas.
Analisar	ANB	Básico	Identifica partes componentes de um todo e reconhece relações simples entre elas.
	ANI	Intermediário	Diferencia fatos de opiniões e examina padrões ou tendências nos dados.
	ANA	Avançado	Avalia estruturas complexas, identifica causas subjacentes e estabelece relações multifacetadas.
Avaliar	AVB	Básico	Expressa opiniões pessoais com justificativas básicas.
	AVI	Intermediário	Crítica ideias ou trabalhos com base em critérios estabelecidos, oferecendo feedback construtivo.
	AVA	Avançado	Defende argumentos com base em evidências sólidas, compara diferentes perspectivas e propõe melhorias fundamentadas.
Criar	CRB	Básico	Combina informações conhecidas para formar ideias simples com orientação.
	CRI	Intermediário	Desenvolve soluções originais ou produtos em contextos conhecidos.
	CRA	Avançado	Concebe propostas inovadoras para problemas complexos, demonstrando alta criatividade e originalidade.

Fonte: elaborado pelos autores.

Ao utilizar esta taxonomia, o time docente pode:

- Definir objetivos de aprendizagem claros para cada LBL;
- Selecionar materiais de estudo e atividades de instrução que correspondam ao nível cognitivo desejado;
- Desenvolver avaliações que reflitam adequadamente o nível de compreensão esperado;
- Definir artefatos de desenvolvimento de projeto aderentes ao nível cognitivo desejado e garantir uma progressão adequada do conhecimento ao longo do curso.

Por exemplo, no módulo M5, que aborda o projeto de sistemas usando plataformas distribuídas (redes, internet e cloud computing), os objetivos de aprendizagem para Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF) são estabelecidos nos níveis "Compreender Básico" (COB) e "Aplicar Básico" (APB).

Esta abordagem permite uma avaliação mais precisa e direcionada do progresso dos alunos, assegurando que eles desenvolvam habilidades cognitivas em todos os níveis, desde a memorização básica até a criação de soluções inovadoras.

3.2 Dimensões de aprendizado baseadas em visões arquiteturais de sistemas

Esta proposta se baseia em uma adaptação de normas internacionais reconhecidas na área de arquitetura de sistemas e software (Pressman, 2021). A ISO/IEC 10746, também conhecida como RM-ODP (Reference Model of Open Distributed Processing), fornece um *framework* para a especificação de sistemas distribuídos. Adaptando os princípios dessas normas para o contexto educacional, propomos cinco dimensões de aprendizado que integram tanto *hard skills* quanto *soft skills*, os desafios multifacetados da indústria de software moderna:

- Visão de Negócio e Business Drivers (BD): Foca na arquitetura de um sistema distribuído alinhada com os direcionadores de negócio. *Hard skills*: Assimilação dos de uma arquitetura controlada, alinhada com os *business drivers*. *Soft skills*: Capacidade de interagir, negociar com pessoas não técnicas com poder decisório.
- Visão de Requisitos Funcionais (RF): Aborda funcionalidade. *Hard skills*: Conectar com pessoas e produtos. *Soft skills*: Negociação dos melhores fluxos de interação.
- Visão de Requisitos Não Funcionais (RNF): Foca pilares da plataforma, incluindo usabilidade, disponibilidade e segurança. *Hard skills*: Projetar sistemas robustos "by design". *Soft skills*: Habilidade de envolver *stakeholders* técnicos e de negócio.
- Visão de Engenharia: Decisões arquiteturais com valor ao negócio. *Hard skills*: implementar táticas arquiteturais. *Soft skills*: decidir por mecanismos arquiteturais.
- Visão de Tecnologia: Ferramentas, plataformas, componentes e serviços. *Hard skills*: integração de tecnologias. *Soft skills*: obter recursos de tecnologias.

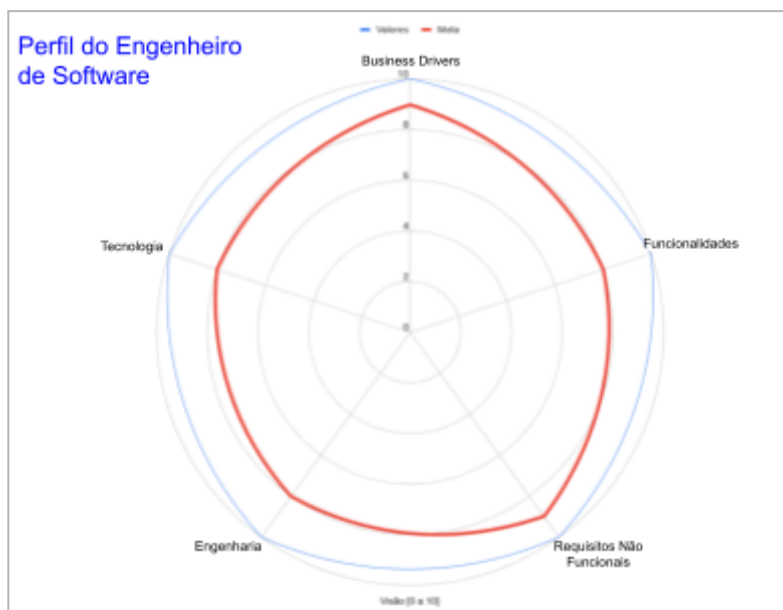
A evolução do aluno em aprendizado sobre sistemas digitais com arquitetura suficiente para atender às necessidades de negócio, tem como meta alcançar uma capacidade em: analisar, avaliar e criar soluções que suportam de maneira controlada as demandas e os riscos que podem afetar a robustez de processamento. O gráfico de radar apresentado na Figura 1 mostra as dimensões refletindo as visões arquiteturais citadas nos itens anteriores, com os critérios atribuídos para cada uma das dimensões conforme ilustrado na Tabela 2.

Tabela 2 - Critérios de pontuação de conhecimento das visões arquiteturais

Visão de Arquitetura	Valores	Meta
Business Drivers	10	9
Funcionalidade	10	8
Req. Não Funcionais	10	9
Engenharia	10	8
Tecnologia	10	8

Fonte: Os autores.

Figura 1 – Gráfico de radar com visões de arquitetura (tabela 2) e metas de formação de engenharia Sw.



Fonte: elaborado pelos autores.

A proficiência e as metas de aprendizado são destacadas desde o conhecer até a fluência em analisar, avaliar e criar soluções de sistemas, expandindo os conceitos associados a cada uma das visões arquiteturais. As tabelas 3 a 7 mostram os critérios de aprendizagem para cada dimensão da arquitetura e justificam o gráfico da Figura 1.

Tabela 3 - Critérios de aprendizagem da Visão Business Drivers.

0~2: Entender e explicar um fluxo de processo de negócio, incluindo a parte a ser digitalizado por sistemas;
2~4: Detectar como o digital adiciona valor para o negócio;
4~6: Analisar volume, regras de negócio, usabilidade como elementos da solução digital;
6~8: Criar os requisitos funcionais e não funcionais combinando os itens de negócio, incluindo os riscos e não riscos e tradeoff, além da adição de valor digital;
8~10: O engenheiro sabe como transformar os itens de negócio como decisão arquitetural controlados digitalmente (cyber), ou seja com mecanismos para detectar, resistir, recuperar e rastrear os eventos que trazem a robustez do sistema, controlando os riscos de instabilidade.

Fonte: elaborado pelos autores.

Tabela 4 - Critérios de aprendizagem da Visão Funcionalidades.

0~2: Entender uma tela e botão, conforme especificado e codificar as telas que implementa esta funcionalidade de acordo com a tecnologia disponibilizada;
2~4: Entende a modelagem: negócio, regra de negócio, fluxo de sucesso, fluxo de exceções;
4~6: Consegue analisar a qualidade com base em revisão estática e dinâmica - massa de testes e cobertura de testes;
6~8: Avalia a evolução e qualidade das funcionalidades aplicando conceitos de gestão de configuração para códigos e documentos e controle de qualidade;
8~10: A suficiência em conhecimento advém do controle de qualidade baseada em código, no qual a documentação é construída como ativo de software - códigos e massa de testes automatizados, com recursos de regressão para aferição da extensão e impactos de mudanças.

Fonte: elaborado pelos autores.

Tabela 5 - Critérios de aprendizagem da Visão Requisitos Não Funcionais.

0~2: Entender as integrações internas e externas: códigos e componentes internos e terceirizados, controlando a evolução de versões de sistemas operacionais e produtos;
2~4: Controlar o desempenho e robustez dos requisitos funcionais por testes não funcionais e integrações de serviços internos e externos;
4~6: Avaliar as táticas arquiteturais e mecanismos não funcionais - medição de requisitos não funcionais;
6~8: Avaliação ATAM da arquitetura e revisão da arquitetura de sistemas (mapa de riscos e controle de qualidade);
8~10: Gestão da qualidade da arquitetura e tradeoffs, por automação com foco em negócios (Embarcar mecanismos de avaliação da arquitetura, baseada nos business drivers);

Fonte: elaborado pelos autores.

Tabela 6 - Critérios de aprendizagem da Visão Engenharia/Mecanismos.

0~2: Testar, documentar e validar as integrações, com abrangências internas, externas;
2~4: Idem anterior para os requisitos não funcionais críticos;
4~6: Medir as táticas arquiteturais e mecanismos não funcionais - Estático e dinâmico;
6~8: Avaliação ATAM da arquitetura e revisão da arquitetura de sistemas (mapa de riscos e controle de qualidade) - focando os códigos (revisão e testes);
8~10: Gestão da qualidade da arquitetura e tradeoffs, por automação com foco em negócios, embarcando mecanismos de avaliação da arquitetura e gestão de configuração, com possibilidades de uso de IA e Machine Learning para suportar os processos de operação e de desenvolvimento (MLOPS e DEVSECOPS);

Fonte: elaborado pelos autores.

Tabela 7 - Critérios de aprendizagem da Visão Tecnologia.

0~2: Testes gerados por equipes internas e terceiras/fornecedores;
2~4: Revisão da qualidade de códigos internos e terceirizados (Gestão configuração);
4~6: Garantir a gestão de ciclo de vida dos componentes e demais dependências;
6~8: Testes e gestão de configuração das mudanças com esteira DEVSECOPS, aplicando o controle de qualidade no processo e no produto;
8~10: Lidar com o aging de tecnologias com integrações de inovações com legado, controlando a diversidade tecnológica de ferramentas, linguagens, plataformas e integrações de hardware software e serviços externos, com foco nos riscos de mudanças e no controle de configuração.

Fonte: elaborado pelos autores.

4 PROPOSTA DE EVOLUÇÃO SISTEMÁTICA DE UM MÓDULO PBL

Um módulo PBL, estabelecido e definido de acordo com as Diretrizes Curriculares Nacionais (DCN/MEC), contém módulos de aprendizagem denominados LBL (Learning Backlog) com pacote de objetos de aprendizagem, incluindo artefatos de projetos, instruções de aula em salas invertidas, auto-estudos com referências científicas ou com práticas industriais, além de exercícios conceituais e práticos. O método (ver figura 2) é centrado em três questões fundamentais:

Questão 1) **Como realizar a evolução sistemática dos módulos de aprendizagem baseados em projetos (LBLs)?**

Para garantir a revisão eficaz dos LBLs, buscamos estabelecer um processo de avaliação repetível e objetivo: Implementar procedimentos padronizados para avaliar a situação atual e identificar melhorias; Utilizar critérios para interpretações objetivas; Empregar referenciais científicos nas decisões; e Reduzir a dependência de ferramentas tecnológicas específicas, para fugir da obsolescência decorrente da dinâmica da indústria.

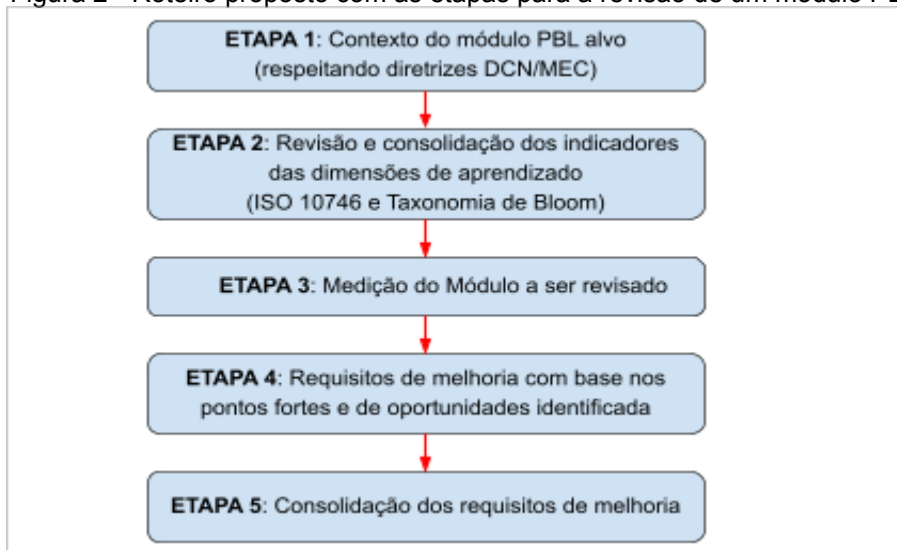
Questão 2) **Quais aspectos da aprendizagem devem ser utilizados como base para compor os requisitos de revisão?**

A revisão dos LBLs é fundamentada em dois pilares principais, orientados por indicadores que facilitem as análises e decisões: A Taxonomia de Bloom, que abrange desde o conhecimento básico até a capacidade de aplicação, avaliação e criação; e As dimensões arquiteturais de um sistema resiliente e robusto, alinhadas com critérios de engenharia, muito mais que as práticas tecnológicas de ferramentas efêmeras nos ciclos de evolução industrial.

Questão 3) **Como (re)configurar os conceitos e práticas sem prejuízo das disciplinas definidas nas DCN/MEC?**

O diferencial da estrutura modular por LBL é a "componentização" do ensino, uma abordagem inovadora do Inteli. Esta abordagem flexível possibilita a reconfiguração dos pacotes de aprendizagem, otimizando o processo educacional sem comprometer os requisitos curriculares essenciais: Ajustar componentes com conteúdos de instrução, auto-estudos de conceitos e práticas; Buscar a melhoria contínua da aprendizagem projetada para cada módulo; Manter a conformidade com as disciplinas estabelecidas nas DCN/MEC, respeitando suas dependências.

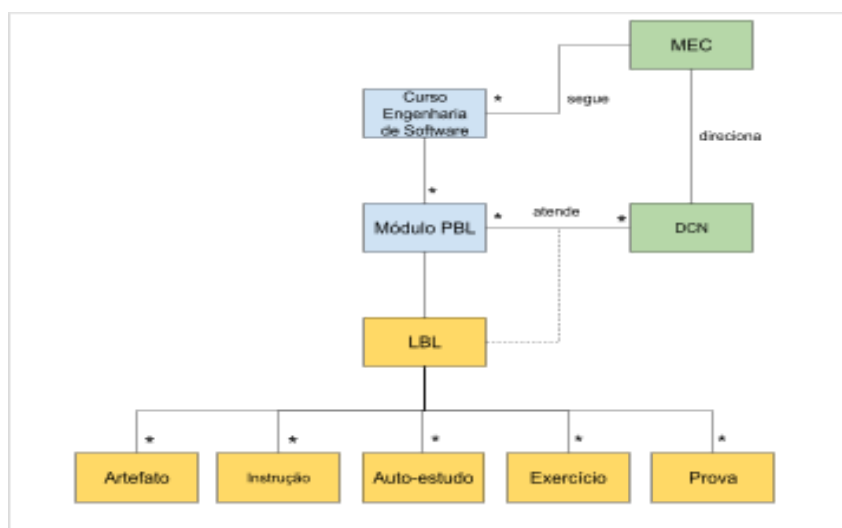
Figura 2 - Roteiro proposto com as etapas para a revisão de um módulo PBL.



Fonte: elaborado pelos autores.

Na Etapa 1, os módulos PBL projetados formam um caminho ao longo do curso de Engenharia de Software para avançar na formação do perfil profissional do aluno. Esse contexto é importante para a revisão do LBL do módulo. Os critérios e indicadores relacionados à arquitetura e taxonomia de Bloom são avaliadas para ver se há alguma necessidade de ajustes, considerando as DCN/MEC na Etapa 2. Já na Etapa 3, revisadas as réguas de qualidade de aprendizagem, medir o módulo alvo da revisão de modo a mapear pontos fortes e oportunidades de melhoria, segundo percepções da equipe de professores (UX, Matemática e Física, Negócios, Liderança e Computação), e avaliações dos alunos. Com o resultado, na Etapa 4 se consolidam os requisitos, destacando os pontos para a melhoria. Para as ações de revisão, na Etapa 5, se consolidam os requisitos de melhoria, priorizadas e selecionadas para que os professores possam atuar na revisão do módulo, aprimorando as instruções, auto-estudos e exercícios de conceitos e da prática de projetos constantes no LBL. A relação estrutural entre os elementos do PBL (Figura 3) ajuda a justificar o método proposto, com especial destaque da conexão do pacote LBL e seus objetos de aprendizagem e a dependência lógica que garantem *compliance* com o DCN/MEC.

Figura 3 - Baixo acoplamento e alta coesão da LBL em relação aos módulos PBL e DCN/MEC.



Fonte: elaborado pelos autores.

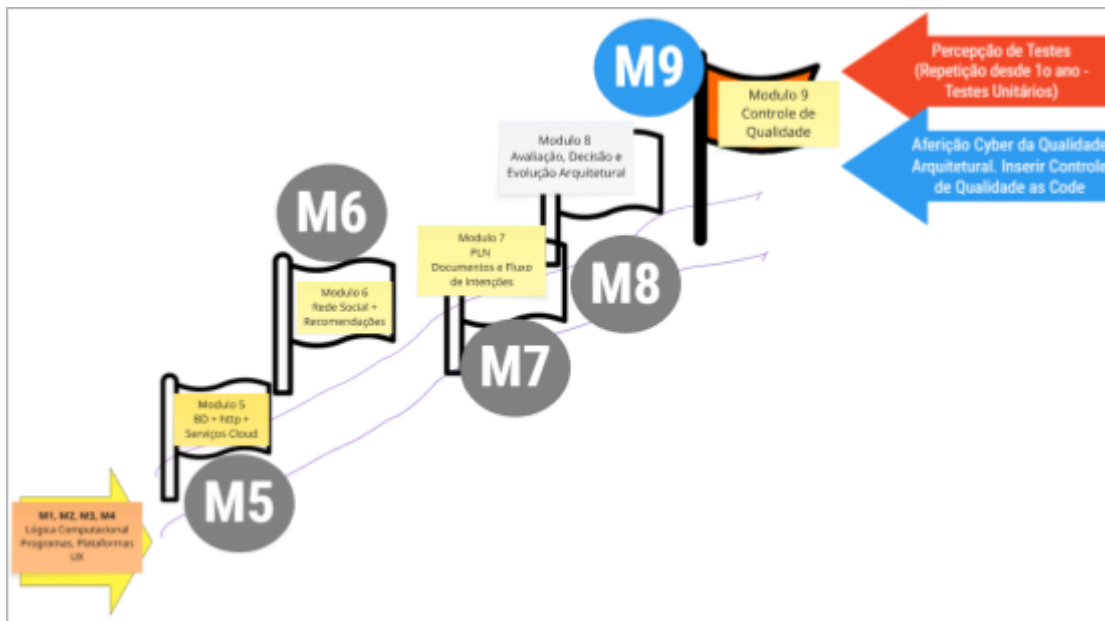
5 EXEMPLO COMENTADO DE APLICAÇÃO DO ROTEIRO

O módulo alvo da revisão é relacionado ao controle de qualidade de um sistema digital, o PBL M9, com o título “Testes automatizados e controle de qualidade de software”. Tem-se os registros documentados da aplicação do roteiro proposto.

5.1 Registros da Etapa 1 - Contexto do módulo PBL alvo da revisão

O esquema na figura 4, ilustra a análise do contexto da revisão. O módulo M9 aparece no segundo ano do curso após módulos básicos do 1o ano (M1 a M4). No M5 inicia-se Engenharia de Software com sistemas distribuídos e integração. No M6, a configuração de uma rede neural para recomendações. No M7, projetos com interação por linguagem natural com áudios, textos livres e vídeos mapeados em comandos e classificação. Aspectos arquiteturais são do M8, onde requisitos não funcionais de segurança, disponibilidade, tolerância a falhas são implementados. O M9, alvo da revisão, avança no controle da qualidade da arquitetura.

Figura 4 - Exemplo de aplicação do método de revisão sobre a M9.



Fonte: elaborado pelos autores.

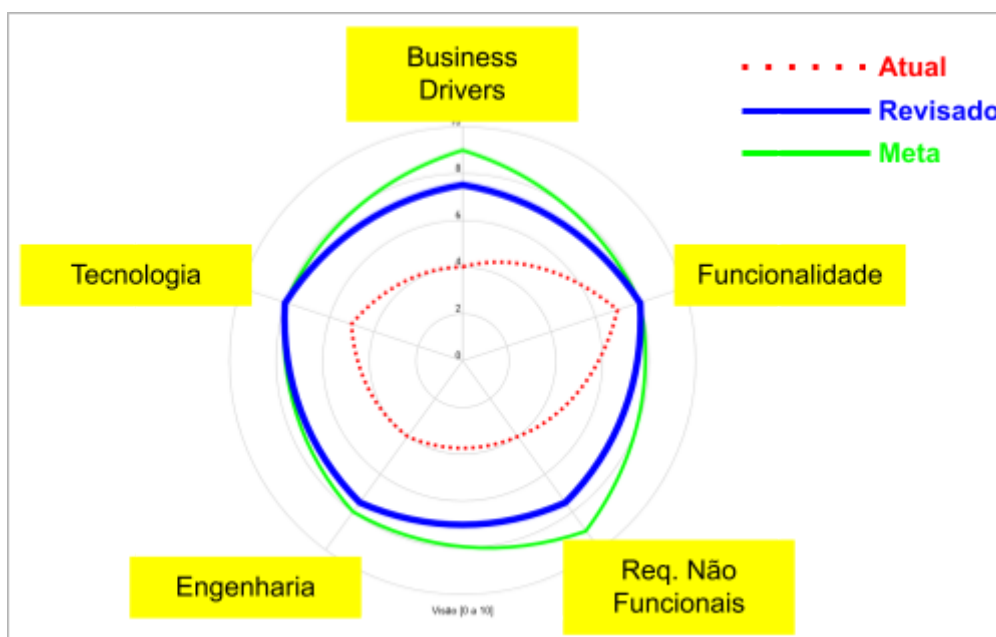
5.2 Registros da Etapa 2 - Revisão e consolidação dos indicadores das dimensões de aprendizado (ISO 10746 e Taxonomia de Bloom)

Os critérios estabelecidos no item 3.2 foram validados e aplicados para esta revisão e continuam válidos para os registros desta etapa. Ver a medição na Figura 5.

5.3 Registros da Etapa 3 - Medição do Módulo a ser revisado

Com base nestes critérios, as medições junto à equipe técnica e avaliações dos alunos, temos a seguinte medição, como mostrado na figura 5. Observe-se que o módulo atual tem foco em funcionalidades e pouca intensidade nos requisitos de arquitetura.

Figura 5 - Medição da situação atual (vermelho) e as lacunas com a meta (verde).



Fonte: elaborado pelos autores.

Ou seja, os pontos de melhoria incluem business drivers, requisitos não funcionais, engenharia e tecnologia, apesar de atenderem satisfatoriamente aos requisitos requisitos funcionais. O desdobramento destes “gaps” levaram aos requisitos direcionadores da revisão, descrito a seguir.

5.4 Registros da Etapa 4 - Requisitos da Revisão

A partir das lacunas de aprendizagem medidas na Etapa 3, a revisão deve ampliar o alcance da aprendizagem do LBL nos seguintes quesitos, como indicado na Tabela 8.

Tabela 8 - Direcionado por Bloom, a avaliação do atual e os itens da revisão.

BLOOM	Percepção do ATUAL	Requisitos da REVISÃO
Memorizar	Não entende testes não funcionais e nem CQ como código.	Consolidar controle de arquitetura as code, iniciado no M8.
Compreender	Não compreende controle de qualidade cibernético.	Como modelar as decisões arquiteturais de CQ as a code.
Aplicar	Confusão de testes como controle de qualidade.	Avaliação arquitetural. Testes não funcionais de mecanismos de CQ.
Analisar	Fluxo de testes unitários e de integração são insuficientes.	Medir e analisar CQ pelas decisões arquiteturais.
Avaliar	Não sabe avaliar a arquitetura em termos de CQ.	Avaliar e identificar GAPs de melhoria, baseado em método.
Criar	Limitação em criar outros projetos.	Criar projetos com arquitetura e CQ as code.

Fonte: elaborado pelos autores.

Em resumo, dos requisitos para aprimoramento do módulo M9, exemplo neste texto, as visões críticas de uma arquitetura que são as do Business Drivers, dos Requisitos não Funcionais, os mecanismos de Engenharia e as plataformas de Tecnologia devem ser ampliadas com o conceito de controle de qualidade, na forma de aferição, na forma de testes e na forma de documentação das restrições como código.

5.5 Registros da Etapa 5 - Consolidação dos requisitos de melhoria

Na etapa 5, conforme os requisitos identificados na Etapa 4, se consolidam as mudanças decididas para a revisão. Na tabela 9 indica-se ampliação da revisão em relação ao atual.

Tabela 9 - Comparação dos requisitos da M9 antes e depois da revisão.

	Visão 1 - Business Drivers	Visão 2 - Requisitos Funcionais	Visão 3 - Requisitos Não.Funcionais	Visão 4 - Engenharia	Visão 5 - Tecnologia
M9 Atual	Foco em arquitetura C4 (implementação)	Boas práticas de arquitetura e testes.	Métricas de Software e performance de código.	Ferramentas de monitoração.	Configuration manager e releases frequentes.
M9 Revisado	Foco no controle dos direcionadores de negócio, como código. O aluno é capaz de avaliar e criar mecanismos de controle de qualidade cyber para os fluxos críticos de negócio.	Aferir resiliência do sistema em disponibilidade, desempenho e segurança. O aluno deve ser capaz de avaliar se os pilares funcionais estão sustentados por pilares não funcionais.	Aferição estática e dinâmica automatizada de qualidade com táticas (resistir, detectar e recuperar). O aluno deve saber avaliar e criar mecanismos de controle dos requisitos.	Evidenciar que os mecanismos de integração de controle de qualidade as code. O aluno deve avaliar o sistema, com base nos controle automatizado de qualidade	Evidenciar que a tecnologia implementa rastros da operação e sustentação. O aluno deve criar mecanismos cyber para resistir, detectar e recuperar de riscos presentes.

Fonte: elaborado pelos autores.

Da tabela 9, além da uniformização focada em código, as *soft skills* estão nas atitudes de engajamento dos *stakeholders* pela importância em qualidade: A visão dos negócios em volumetria, acessos simultâneos e disponibilidade; A de requisitos, o engajamento de stakeholders em conjunto para criar códigos que avaliam os níveis de serviços e regras; A de engenharia para as integrações com aferição dos níveis de serviços contratados; e por fim a de Tecnologia, com controle na sustentação operacional e na evolução por mudanças. Observa-se ampliação da capacidade em análise crítica e implementação de controle de qualidade na aprendizagem estabelecida para o módulo M9 revisada.

6 RESULTADOS OBTIDOS

A aplicação do método para evolução sistemática de módulos PBL resultou em avanços significativos na revisão dos Learning Backlogs (LBLs) do curso de Engenharia de Software. A sistemática demonstrou viabilidade prática e coerência conceitual, permitindo maior controle e clareza, tais como:

- Sistemática de Revisão de LBL: A proposta permite que diferentes times docentes avaliem e aprimorem os LBLs de forma uniforme, reduzindo a subjetividade.
- Alinhamento entre competências e aprendizagem: A combinação da taxonomia de Bloom com visões arquiteturais estabeleceu critérios com foco em capacitação.
- Modularidade pedagógica: Elementos LBLs conectados a objetivos pedagógicos, facilita melhorias contínuas com a devida adequação e compliance com o MEC.
- Clareza nos requisitos de melhoria: O método permitiu transformar a percepção dos envolvidos em requisitos objetivos de evolução, com base em critérios técnicos e pedagógicos.
- Base para automatização futura: A estrutura sistemática, baseada em indicadores observáveis e métricas cognitivas e arquiteturais, fornece um alicerce sólido para o desenvolvimento de ferramentas automatizadas de apoio à revisão curricular.

7 CONSIDERAÇÕES FINAIS

Neste estudo, foi criado um método sistematizado para melhorar os módulos de Aprendizagem Baseada em Projetos (PBL) em programas de Engenharia de Software. Ele combina a taxonomia de Bloom com perspectivas arquiteturais baseadas em padrões globais. Essa abordagem permite a avaliação objetiva dos PBLs, resultando em uma formação mais sólida e alinhada com as Diretrizes Curriculares Nacionais (DCN/MEC). Ao usar elementos arquiteturais como base para o desenvolvimento de habilidades técnicas e socioemocionais, oferece-se uma visão estruturada do percurso educacional dos alunos em direção à consolidação de seu perfil profissional. Os resultados indicam que essa metodologia promove consistência pedagógica e facilita a identificação de áreas para melhorias nos módulos. Uma contribuição significativa foi a criação de uma ferramenta prática e replicável para auxiliar na tomada de decisões sobre a atualização dos objetos de aprendizado sem colocar em risco o alinhamento com as diretrizes curriculares do MEC, porém com flexibilidade para a busca de qualidade do ensino baseado em projetos. Como próximas evoluções planeja-se automatizar partes da metodologia a partir dos dados e percepções coletadas das aplicações dos módulos revisados junto aos estudantes e empresas parceiras.

AGRADECIMENTOS

Os autores agradecem à estrutura executiva e de coordenação do Curso de Engenharia de Software do Instituto INTELI pelo o apoio nos estudos, discussões e revisões dos conteúdos dos módulos PBL, sempre buscando o alinhamento junto ao MEC e observando os resultados das atividades de aprendizagem dos alunos e interações com empresas parceiras.

Os autores também agradecem o acolhimento e o apoio da equipe de professores de pós-graduação no ensino superior em PBL, pela busca incessante de desafios para a aprendizagem e aprimoramento das técnicas em salas de aula invertida.

REFERÊNCIAS

APEANTI, Wilson Osafo; ESSEL, Daniel Danso. Learning Computer Programming Using Project-Based Collaborative Learning: Students' Experiences, Challenges, and Outcomes. August. International Journal for Innovation Education and Research 9(8):191-207, 2021.

ARAKAKI, Reginaldo *et al.* Avaliação do oferecimento a distância de laboratório de eletrônica digital por meio de objetivos de aprendizagem e métricas do AVA. In: ANAIS do XLIX Congresso Brasileiro de Educação em Engenharia, 2021.

BARROWS, Howard. S. Problem-Based Learning in Medicine and Beyond: A Brief Overview. New Directions for Teaching and Learning, 3-12, 1996;

CUGNASCA, Paulo Sérgio *et al.* O Papel do Laboratório de Eletrônica Digital como Disciplina Integradora de Competências para o Desenvolvimento de Projetos de Engenharia. In: 52º Congresso Brasileiro de Educação em Engenharia, Vitória, 2024.

FELDER, R.; SOLOMAN, B. Index of Learning Styles (ILS). Disponível em: <https://learningstyles.webtools.ncsu.edu/> Acesso em: 01/06/2025.

FIORAVANTE, M.L.; SENA, B.; PASCHOAL, L.N. Integrating project based learning and project management for software engineering teaching: An experience report. Proceedings of the 49th ACM Technical Symposium on Computer Science Education. Páginas 806-811. Editora ACM, 2018.

HABIMORAD, Maira; 2024. Acessar o site do Instituto Inteli. <https://www.inteli.edu.br> e solicitar **RELATÓRIO-ANUAL-2024-22082024-FINAL-DIGITAL.pdf**, Acessado na Internet em Janeiro 2025.

HAYASHI, V. *et al.* Laboratório Remoto para o Ensino de Engenharia. In: S WORKSHOPS DO CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (WCBIE), 9. , 2020, Online. Porto Alegre: Sociedade Brasileira de Computação, p. 187-194, 2020.

HAYASHI, V. T. *et al.* **Laboratório Virtual com Dados Reais**. In: SIMPÓSIO BRASILEIRO DE EDUCAÇÃO EM COMPUTAÇÃO (EDUCOMP), 1. , 2021, On-line. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 296-304. DOI: <https://doi.org/10.5753/educomp.2021.14497>;

HAYASHI, Victor Takashi *et al.* Implementation of PjBL With Remote Lab Enhances the Professional Skills of Engineering Students. August 2023. IEEE Transactions on Education PP(99):1-10, 2023.

INTELI PPC. Acessar <https://www.inteli.edu.br/engenharia-de-software>. Projeto Pedagógico do Curso de Bacharelado em Engenharia de Software. 2024. Acesso em Janeiro de 2025.

ISO/IEC 12207:2008 Systems and software engineering – Software life cycle processes http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=43447;

ISO/IEC 25010:2011 - Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models
http://www.iso.org/iso/catalogue_detail.htm?csnumber=35733;

ISO/IEC/IEEE 42010:2022. Systems and software engineering — Architecture description
<https://www.iso.org/standard/74393.html>;

ISO/IEC 10746-2:2009 Information technology — Open distributed processing — Reference model: Foundations;

MARKHAM, T.; LARMER, J.; RAVITZ, J.; Aprendizagem Baseada em Projetos, Artmed Editora S/A, Porto Alegre, 2008.

PRESSMAN, Roger; MAXIM, Bruce; ARAKAKI, Reginaldo, (Revisão Técnica) Engenharia de Software: Uma Abordagem Profissional, Makron Book, January 2021.

ZHANG, M.; HU, W., (2024) Application of PBL combined with CBL teaching method in clinical teaching of vascular surgery. PLoS ONE 19(8), 2024.

SYSTEMATIC OPTIMIZING OF SOFTWARE ENGINEERING PBL (PROJECT BASED LEARNING) MODULE: A LEARNING GOALS-BASED METHOD USING ARCHITECTURE VIEWPOINTS

Abstract: *This work details a methodology to review the learning modules backlog named LBL based on Project Based Learning (PBL), aligned with curricular rules and directions of MEC (Ministério de Educação) and PPC (Projeto Pedagógico do Curso) of Software Engineering Bachelor, considering industry professional profiles expected in terms of capacity, ability, and knowledge to build and maintain digital software platforms. This research drew on the experiences and disciplines of business, leadership, computation, mathematics, physics, and user experience experts. Scientific architecture references include ISO/IEC 25010, ISO/IEC 12207, and ISO/IEC 10746, which encompass five viewpoints of a digital system: Business Drivers, Functional and Non-Functional Requirements, Engineering Solutions, and Technology Tools.*

Such dimensions of approach review are compatible with engineering aspects of modern and complex digital platforms that lead with the human body, transportation devices, and connected cities all digitally integrated.

Keywords: *Curriculum Alignment, Bloom's Taxonomy, Architecture Viewpoints, Modular Design, Learning Assessment, ISO/IEC 42010, Soft and Hard Skills, LBL (Learning Backlog), Learning Objects, Quality as Code, Document as Code, Project Based Learning.*

