



## **API – APRENDIZAGEM POR PROJETOS INTEGRADOS**

### **GUIA DE AVALIAÇÃO**

**Revisão: 2025-2**

#### **Comissão Editorial:**

Luiz Antonio Tozi, José Walmir Gonçalves Duque, Eduardo Sakaue,  
Reinaldo Gen Ichiro Arakaki, Fabiano Sabha Walczak, Emanuel Mineda Carneiro

#### **Redator Responsável:**

José Walmir Gonçalves Duque



## API – APRENDIZAGEM POR PROJETOS INTEGRADOS

### GUIA DE AVALIAÇÃO

Revisão: 2025-2

#### Sumário

1. Introdução.....	3
2. Avaliação – Orientações Gerais.....	4
2.1. Quadro Geral de Avaliação .....	4
2.2. Aplicação das Notas nas Disciplinas da Tríade .....	5
2.3. Outras Observações Importantes.....	5
3. Itens de Avaliação.....	6
Processo .....	6
GitHub .....	8
Requisitos.....	10
Soft Skills.....	12
Anexos .....	13

## 1. Introdução

### *O que é a API?*

A **API (Aprendizagem por Projetos Integrados)** é uma Metodologia de Ensino desenvolvida pela Fatec São José dos Campos e implantada desde o segundo semestre de 2019, inicialmente nos cursos da área de Computação. Atualmente já existem variações da metodologia sendo desenvolvidas em outros cursos.

A API foi proposta ante um cenário desafiador para empresas que, ao contratar seus colaboradores, percebem que cada vez mais não somente as hard skills são suficientes, mas também as soft skills.

Portanto, os pilares da metodologia foram definidos a partir de uma leitura do ensino superior tecnológico no contexto dos processos mais amplos de digitalização da sociedade, conforme segue:

- **Real Problem Based Learning (rPBL):** foi empregada uma variação do rPBL na qual são prospectados desafios reais junto às empresas ou organizações (denominamos parceiros acadêmicos), elaborados de acordo com as necessidades didático-pedagógicas de cada turma.

- **Validação Externa:** os parceiros acadêmicos tem papel importante na realização do crosscheck (validação cruzada) - desde a concepção do desafio dos projetos até um certo nível de acompanhamento das entregas.

- **Mindset Ágil (Agile):** foi agregada na metodologia API elementos de Gestão Ágil de Projetos, inspirados no framework Scrum, incorporando alguns papéis e cerimônias adaptadas para a nossa realidade acadêmica.

Vale ressaltar que API não é uma disciplina, mas uma Metodologia de Ensino que é parte importante dos cursos!

### *Cursar API é obrigatório?*

Sim, é obrigatório, pois:

- A **API** consta formalmente nos Projetos Pedagógicos de Curso (PPCs) dos três cursos do Eixo TI: ADS, BD e DSM;
- A **API** nos dá suporte formal e evidências para o atendimento da Curricularização da Extensão, obrigação legal para Instituições de Ensino Superior no Brasil, estabelecida pelo Conselho Nacional de Educação pela Resolução Nº 07 de 18 de Dezembro de 2018.

Ademais, é **importante destacar que acreditamos muito nos resultados de aprendizado trazidos pela API para nossos alunos** – ao final do seu curso, cada aluno tem o privilégio e o diferencial de ter um Portfólio de Projetos reais em seu GitHub!

*Vamos mergulhar nessa grande oportunidade de aprendizado?*

## 2. Avaliação – Orientações Gerais

### 2.1. Quadro Geral de Avaliação

Categoria	ID#	Item Avaliado	Peso
Processo	P.1	Backlog de Produto	10%
Processo	P.2	Backlog de Sprint	10%
Processo	P.3	Alocação de Tarefas, com base no DoR	10%
GitHub	G.1	Documentação no GitHub	5%
GitHub	G.2	Estratégia de Branch	5%
GitHub	G.3	Padrão de Mensagem dos Commits	5%
Requisitos	R.1	Review - Apresentação	15%
Requisitos	R.2	Conformidade Técnica - Requisitos Funcionais e Não Funcionais	20%
Soft Skills	SOFT	PACER	20%

Cada **Item Avaliado**, disposto no quadro acima, terá uma avaliação na **Escala de 1 a 4**, que corresponderá a notas de **0.0 a 10.0**, conforme tabela a seguir:

1	Não atendeu	0,0
2	Regular	4,0
3	Bom	7,0
4	Muito bom	10,0

## 2.2. Aplicação das Notas nas Disciplinas da Tríade

Uma vez obtidas as notas da API, as disciplinas da Tríade deverão aplicar as notas conforme os pesos definidos a seguir:

Disciplina	Peso
Disciplina Focal (do M2)	Mínimo <b>70%</b> ( <b>60% API + 10% Feira de Soluções</b> )
Disciplinas Envolvidas (do P2 e Técnica)	Mínimo <b>50%</b>

**Nota mínima de uma Disciplina da Tríade para uso da nota da API:** caso o aluno não obtenha o conceito mínimo de **6,0** de nota da disciplina (Focal ou Envolvida), a nota do API será descartada.

## 2.3. Outras Observações Importantes

### a) Não há Reprovação na API

Se a nota final do API não for suficiente para aprovação na disciplina FOCAL, o professor da disciplina pode aplicar uma nova avaliação como EXAME/RECUPERAÇÃO, Modelo Tradicional ou Projeto, e a nova nota impactará apenas na disciplina FOCAL. As demais disciplinas ENVOLVIDAS devem utilizar a primeira nota final do API, antes desta RECUPERAÇÃO.

### b) Sobre abandonar uma Sprint

A apresentação em todas as Sprints é obrigatória - em caso de uma equipe optar por não apresentar uma Sprint (abandonar uma Sprint), a Equipe terá a nota do seu projeto zerado.

### 3. Itens de Avaliação

#### Processo

Na categoria "Processo" são observados Itens de Avaliação (critérios) que refletem a atuação do Scrum Master, Product Owner (PO) e DevTeam, levando em conta a forma e o conteúdo das evidências aqui estabelecidas.

#### P.1. Backlog de Produto

Avaliador: P2

Avaliado: PO (principal responsável) e demais membros da equipe

Peso: **10%**

**Referências:** Anexo I – Template de Backlog de Produto; Orientações Adicionais em "Guia de Artefatos de Requisitos – Métodos Ágeis"

Avaliação – Escala:

1	Não atendeu
2	Preparação antes da Sprint 1, Adequação ao Template
3	Preparação antes da Sprint 1, Adequação ao Template, <b>Priorização por Valor, Interação com Parceiro</b>
4	Preparação antes da Sprint 1, Adequação ao Template, Priorização por Valor, Interação com Parceiro, <b>Explicitação de Valor do Negócio nas User Stories</b>

## P.2. Backlog de Sprint

Avaliador: P2

Avaliado: PO (principal responsável) e demais membros da equipe

Peso: **10%**

**Referências:** Anexo II - Template de Backlog de Sprint; Orientações Adicionais em "Guia de Artefatos de Requisitos – Métodos Ágeis"

Avaliação – Escala:

1	Não atendeu
2	Preparação antes da Sprint
3	Preparação antes da Sprint, Precisa conter as User Stories de maior Prioridade, Definição de Meta (imutável)
4	Preparação antes da Sprint, Precisa conter as User Stories de maior Prioridade, Definição de Meta (imutável), Cumprimento de DoR

## P.3. Alocação de Tarefas, com base no DoR

Avaliador: M2

Avaliado: Master (principal responsável) e demais membros da equipe

Peso: **10%**

**Referências:** Anexo III - Template para Checklist de DoR; Orientações Adicionais em "Guia de Artefatos de Requisitos – Métodos Ágeis"

Avaliação – Escala:

1	Não atendeu
2	Distribuição e Classificação de Tarefas proporcional aos membros
3	Criação de Tarefas com tamanho adequado (duração prevista menor que 8 horas), Distribuição e Classificação de Tarefas proporcional aos membros, Alocação de uma tarefa em progresso por vez, Estimativa de Data de entrega de Tarefa
4	Criação de Tarefas com tamanho adequado (duração prevista menor que 8 horas), Distribuição e Classificação de Tarefas proporcional aos membros, Alocação de uma tarefa em progresso por vez, Estimativa de Data de entrega de Tarefa, Atualização de Burndown com Progresso

Orientações:

- Sobre **Burndown da Sprint**: importante trabalhar os conceitos envolvidos tais como o acompanhamento da evolução do cumprimento das tarefas e transparência. Exemplos de ferramentas possíveis: Jira, Template em Excel ou outras

Faculdade de Tecnologia de São José dos Campos - "Professor Jessen Vidal"

ferramentas que consigam dar suporte aos conceitos envolvidos. Se o professor não definir (e ensinar), o aluno fica à vontade para escolher.

## GitHub

Na categoria "GitHub" são observados Itens de Avaliação (critérios) que refletem o uso do GitHub como ferramenta de repositório para armazenar a documentação do projeto. Também se observa nesta categoria o uso do GitHub na dinâmica de interação da equipe e na implementação de suas estratégias de versionamento de código, assim, favorecendo o trabalho colaborativo.

### G.1. Documentação no GitHub

Avaliador: P2; Manual de Instalação (Professor M2, P2 e/ou Disciplina Envolvida)

Avaliado: Todos os Membros da Equipe

Peso: **5%**

Referências: Anexo IV – Itens para compor o README do GitHub; Orientações Adicionais em "Guia do GitHub"

Avaliação – Escala:

1	Não atendeu
2	Membros da Equipe com respectivos papéis, Descrição do desafio, Título do projeto, Backlog do produto, Pasta de documentação
3	Membros da Equipe com respectivos papéis, Descrição do desafio, Título do projeto, Backlog do produto, Pasta de documentação, Tecnologias utilizadas, Descritivo da Sprint (links para pasta de documentação com DoD/DoR da Sprint e vídeo de incremento no Youtube)
4	Membros da Equipe com respectivos papéis, Descrição do desafio, Título do projeto, Backlog do produto, Pasta de documentação, Tecnologias utilizadas, Descritivo da Sprint (links para pasta de documentação com DoD/DoR da Sprint e vídeo de incremento no Youtube), Manual de instalação, Manual de Usuário

Orientações:

- Minicurso de utilização de Git deve ser ministrado, já no primeiro semestre e reforçado, a cada semestre.

Excelente para aprender criação uso do Git e Branches: [https://learngitbranching.js.org/?locale=pt\\_BR](https://learngitbranching.js.org/?locale=pt_BR)

Exemplos: <https://cursa.com.br/home/course/curso-b%C3%A1sico-de-git/75>



## G.2. Estratégia de Branch

Avaliador: M2

Avaliado: Todos os Membros da Equipe

Peso: 5%

Referências: Orientações Adicionais em "Guia do GitHub"

Avaliação – Escala:

1	Não atendeu
2	Documentação e Justificativa da Estratégia de Branch
3	Documentação e Justificativa da Estratégia de Branch, Aplicação da Estratégia de Branch
4	Documentação e Justificativa da Estratégia de Branch, Aplicação da Estratégia de Branch, Uso de MR (Merge Request), Uso de PR (Pull Request) - indicada pelo Professor Disc. Técnica

## G.3. Padrão de Mensagem dos Commits

Avaliador: M2

Avaliado: Todos os Membros da Equipe

Peso: 5%

Referências: Orientações Adicionais em "Guia do GitHub"

Avaliação – Escala:

1	Não atendeu
2	Definição de Padrão de Mensagem (tipo de alteração, tarefa, breve descrição)
3	Definição de Padrão de Mensagem (tipo de alteração, tarefa, breve descrição), Aplicação do padrão na maioria dos commits
4	Definição de Padrão de Mensagem (tipo de alteração, tarefa, breve descrição), Aplicação do padrão em todos os commits

## Requisitos

Na categoria "Requisitos" são observados Itens de Avaliação (critérios) que refletem o atendimento aos Requisitos Funcionais e Não Funcionais definidos no Desafio proposto, em específico, o que foi elaborado e programado pela Equipe para uma Sprint. Para isso, são avaliadas a Apresentação e suas entregas de Requisitos Funcionais e Não Funcionais implementados (em sua forma e conteúdo). Nesta categoria também é avaliada a conformidade técnica das implementações.

**Importante:** cada requisito funcional pode ser reavaliado em uma nova Sprint, uma vez que o esforço a ele concentrado extrapole a janela de tempo de uma sprint ou, então, o requisito funcional foi incrementado ou corrigido pela equipe - contudo, uma nova avaliação de requisito funcional não substitui a avaliação da sprint anterior.

### R.1. Review – Apresentação

Avaliador: P2

Avaliado: Todos os Membros da Equipe

Peso: **15%**

Referências: Anexo V – Itens para compor uma Apresentação de Sprint

Avaliação – Escala:

1	Não atendeu
2	Conteúdo (Apresentou todas as User Stories previstas e "algo" funcionando)
3	Conteúdo (Apresentou <b>todas as User Stories da Meta do Sprint funcionando</b> e seguindo o template da apresentação), Postura (linguagem, time todo presente, respondeu adequadamente as perguntas), Uso do tempo
4	Conteúdo (Apresentou <b>todas as User Stories previstas funcionando</b> e seguindo o template da apresentação), Postura (linguagem, time todo presente, respondeu adequadamente as perguntas), Uso do tempo

## R.2. Conformidade Técnica – Requisitos Funcionais e Não Funcionais

Avaliador: Disciplina Técnica

Avaliado: Todos os Membros da Equipe

Peso: 20%

Referências: "Conteúdos ministrados pelos professores de Disciplina Técnica"

Avaliação – Escala:

1	Não atendeu
2	Avaliação do professor – Inspeção de Código/Avaliação Qualitativa e Quantitativa das Funcionalidades
3	Avaliação do professor – Inspeção de Código/Avaliação Qualitativa e Quantitativa das Funcionalidades
4	Avaliação do professor – Inspeção de Código/Avaliação Qualitativa e Quantitativa das Funcionalidades

Orientações:

- Entrega de código fonte em funcionamento é obrigatória para todas as turmas a partir do 2º semestre dos cursos do Eixo TI.
- Os critérios de qualidade para avaliação serão definidos por cada professor de disciplina técnica em seu "Code Review".
- Necessariamente deverão ser avaliados os Conhecimentos (incluem as Tecnologias) definidos no documento "Desafio do Parceiro Acadêmico".
- Assim, caberá aos professores das disciplinas envolvidas (técnicas de apoio), em conjunto, definirem a nota que cabe a este critério em uma dada Sprint.

## Soft Skills

Na categoria "Soft Skills" são observados Itens de Avaliação (critérios) que refletem a maturidade socioemocional de cada um dos membros da equipe.

São considerados 4 critérios, reunidos em uma sigla PACER:

**PACER – Proatividade, Autonomia, Colaboração e Entrega de Resultados**

Avaliador: Todos os Membros da Equipe, com a Supervisão do M2

Avaliado: Todos os Membros da Equipe

Peso: **20%**

Referências: "Guia de Soft Skills API"

Avaliação – Escala:

Nível	Proatividade	Autonomia	Colaboração	Entrega de Resultados
<b>1</b>	Reativo	Dirigido	Ausente	Não há entregas
<b>2</b>	Desfocado	Gerido	Cooperativo	Entregas duvidosas
<b>3</b>	Ativo	Treinado	Coordenado	Entregas confiáveis
<b>4</b>	Proativo	Autogestão	Colaborativo	Entregas perfeitas

Orientações:

- **Realocação de alunos (antiga "demissão")** pode ocorrer à partir de "critérios de permanência" definidos pela equipe e sob a Supervisão do Professor M2.
- **Se o aluno atingir Nível baixíssimo em todas os critérios de Soft Skills (Nível 1)**, não será zerada sua nota na Sprint mas apenas no critério Soft Skills. Cabe, portanto, à equipe a aplicação de seus "critérios de permanência" para este membro mal avaliado.

Aplicação da Avaliação: ao término de cada Sprint, sob a Supervisão do Professor M2:

- M2 se reúne com a equipe em separado;
- M2 solicita ao Master da equipe para enviar para todos os membros os nomes dos alunos para usarem para envio de notas;
- M2 explica e discute o critério "Proatividade" e como a equipe acompanha e aplica no seu dia-a-dia;
- Ao sinal do M2, cada aluno da equipe avalia os seus colegas (menos a si mesmo) e envia as notas para o seu Master;
- M2 repete os passos (c) e (d) para os demais critérios (Autonomia, Colaboração e Entrega de Resultados);
- Ao final, o Master envia para o M2 (pelo chat do Teams), uma tabela com os nomes dos alunos e as médias das notas de cada critério;
- Posteriormente: o Master se reúne com a equipe e atualiza os critérios de permanência com os indicadores colhidos – também é o momento de tomada de decisão sobre **realocação de membros!**

## Anexos

### Anexo I – Template de Product Backlog

Rank	Prioridade	User Story	Estimativa	Sprint
1	Alta	Como síndico, quero abrir solicitações de vistoria predial para atender à legislação de meu município	5	1
2	Alta	Como atendente, quero gerenciar as solicitações de vistoria para agendar atendimento	8	1
3	Alta	Como síndico, quero receber o laudo da vistoria predial para anexar ao processo de auditoria da prefeitura	5	1
...	Média	...	...	...
17	Baixa	Como atendente, quero um relatório de aniversários dos funcionários do condomínio que utilizam o sistema para enviar cartões acompanhados de brindes	2	6
18	Alta	Como síndico, quero realizar a abertura dos portões eletrônicos do condomínio através do e-Condos, para que eu possa reaproveitar a experiência dos condôminos com o aplicativo e-Condos e sua adesão a 100% dos moradores fornecendo uma alternativa a <i>tag</i> atual	40	?

## Anexo II – Template de Backlog de Sprint

Capacidade estimada da Equipe por Sprint:	16 Story Points
<b>Meta da Sprint:</b>	User Stories de rank 1 e rank 2 (total de 13 <i>Story Points</i> )
Previsão da Sprint (extras, sem compromisso de entrega):	User Story de rank 3 (5 <i>Story Points</i> )

Rank	Prioridade	User Story	Estimativa	Sprint
1	Alta	Como síndico, quero abrir solicitações de vistoria predial para atender à legislação de meu município	5	1
2	Alta	Como atendente, quero gerenciar as solicitações de vistoria para agendar atendimento	8	1
3	Alta	Como síndico, quero receber o laudo da vistoria predial para anexar ao processo de auditoria da prefeitura	5	1

## Anexo III – Template para Checklist de DoR

### Checklist: (pode ser inserido no JIRA, por exemplo)

Sobre User Stories:

- ☐ Tem título claro, descrição bem definida e objetivo compreendido
- ☐ Tem critérios de aceitação escritos
- ☐ Tem regras de negócio claras
- ☐ Foi estimada pela equipe
- ☐ Sem dependências bloqueadoras
- ☐ Compreensão validada com o time

Sobre artefatos correlatos às User Stories:

- ☐ Design/documentação disponível
- ☐ Regras de negócio detalhadas (texto ou diagrama)
- ☐ Modelo de dados disponível
- ☐ Estratégia de testes definida

## Anexo IV – Itens para compor o README do GitHub

### README (principal)

- Título do Projeto
- Descrição do Desafio (com a dor do Parceiro)
- Backlog de Produto
- Cronograma de evolução do Projeto (de forma visual)
- Tabela descritiva das Sprints com as colunas:
  - . Período da Sprint
  - . Link para Documentação da Sprint
  - . Link para Vídeo no Youtube do Incremento entregue
- Tecnologias utilizadas
- Estrutura do Projeto
- Como executar, usar e testar o projeto
- Link para Pasta de Documentação
  - . Checklist de DoR e DoD
  - . DoR e DoD, em si, por Sprint
  - . Estratégia de Branch
  - . Manual de Usuário
  - . Manual de Instalação
- Equipe (com nome completo, papel, foto, Link para GitHub e Link para LinkedIn)



## Anexo V – Itens para compor uma Apresentação de Sprint

- Slide de Capa com Logos institucionais, Título do Projeto e Nome da Equipe
- Apresentação da Equipe (com nomes completos, papel e foto) e QRCode do Link do GitHub
- Descrição do Desafio (com a dor do Parceiro)
- Backlog de Produto
- Cronograma de evolução do Projeto (de forma visual)
- Backlog da Sprint, evidenciando sua Meta da Sprint
- Tecnologias utilizadas
- Apresentação do Incremento em funcionamento, explicando o atendimento aos Requisitos
- Documentação da Sprint