

Escopo e objetivos — Proposta de Projeto

Os estudantes formarão times de 2 ou 3 integrantes. Cada time **enviará para leite@utfpr.edu.br, até 28/04/2022 23:59**, uma Proposta de Projeto **Web** à escolha do time (site ou aplicativo), na qual deverá constar:

- Integrantes do time (nome e e-mail)
- Endereço do repositório GitHub¹ **público** contendo o *snapshot* **inicial** do trabalho (1 ou mais *commits*)
- Breve descrição das modificações pretendidas (páginas/rotas/funcionalidades a serem introduzidas ou excluídas, problemas a serem sanados etc.), com diagramas, protótipos e outros recursos que auxiliem a compreensão do escopo²
- Pré-requisitos de sistema e tecnologias envolvidas³

¹ O repositório deverá ser gerenciado pela equipe, podendo ser **inicializado em branco** ou a partir de ***fork de projeto*** já existente, por exemplo. Todo o código inicial, independentemente de sua origem, deve ser disponibilizado no repositório no momento do envio da Proposta de Projeto.

² Recomenda-se manter o escopo pequeno, ou seja, não há necessidade de inicializar (ou dar continuidade a) um projeto grande. O objetivo é consolidar as habilidades de produção de código versionado e gerenciamento de projeto colaborativo (paralelo/não-linear); desse modo, o foco da avaliação estará no código produzido pelo time **após** o *snapshot* inicial, e não **nesse** *snapshot* em si. As modificações pretendidas deverão derivar na etapa seguinte atividades **independentes**, conforme detalhado na seção “Desenvolvimento do Projeto”.

³ Em suma, pretende-se evitar que o professor ou o time tenham dificuldades em compilar/executar o código em uma máquina Debian atualizada com configurações de *hardware* modestas. **Recomenda-se** evitar linguagens compiladas/transpiladas ou que gerem binários/*byte-code* em geral, dando preferência para HTML5, CSS3, JavaScript (e outros *scripts*) na medida do possível. Fica **vedado** o uso de tecnologias, ferramentas e componentes que envolvam custo financeiro de qualquer natureza (licenciamento).

Sugestões de códigos iniciais são o “HTML5 Boilerplate” (bem básico), os projetos desenvolvidos nas disciplinas WEB01 (Fundamentos Do Desenvolvimento Web) e WEB02 (Introdução aos Frameworks Angular, React, VueJS e EmberJS), *boilerplates* de *hackaton*, *boilerplates* de Bootstrap, entre outros projetos.

Sugestões de modificações são a inserção de novas páginas, introdução de novos componentes (seções, botões e controles), melhorias de layout (inclusive introdução de responsividade), e criação de novas funcionalidades via JavaScript.

Eventuais adequações de projeto solicitadas pelo professor deverão ser **corrigidas até 01/05/2022 23:59**.

Desenvolvimento do Projeto

Uma vez que o time obtenha aceite da Proposta de Projeto, deverá criar no GitHub:

- Ao menos 4 *issues* referentes ao escopo declarado na Proposta, para equipes de 2 membros; OU
- Ao menos 6 *issues* referentes ao escopo declarado na Proposta, para equipes de 3 membros.

Antes de criar os *issues*, pode ser necessário ativar no GitHub o recurso para o projeto. Cada *issue* deverá constituir uma atividade (tarefa) independente das demais, ou seja, deverá ser possível paralelizar as atividades propostas o máximo possível. Serão **vetados** já na etapa de proposta os projetos para os quais se perceba que as tarefas necessitam ser realizadas predominantemente em série (ou seja, em que haja grande grau de interdependência⁴).

⁴ Em suma, pretende-se evitar que o cronograma de execução fique apertado (com um membro tendo que aguardar o outro concluir uma atividade, para só então poder dar andamento à próxima tarefa), e também evitar que o histórico do projeto fique linear demais (lembrando: um dos objetivos é praticar a produção de código em paralelo, com histórico não-linear).

A descrição de cada *issue* deve ser a mais rica possível, com textos, protótipos, diagramas e outros recursos que ajudem a entender o problema e/ou a solução esperada⁵. **Cada** integrante deverá ser atribuído **a ao menos 2** desses *issues*, ou seja, ficar como responsável principal pelo desenvolvimento técnico da atividade (sem prejuízo para eventuais colaborações entre os membros do time). **Nenhum** integrante deve ser responsável **por menos que 2 issues**.

⁵ **Dica**: muitas vezes, o linguajar de um *issue* não chega a ser técnico, ou seja, não se costuma discutir práticas de codificação, indicar os arquivos e linhas a serem alterados etc. Esse tipo de detalhe fica melhor localizado no(s) *Pull Request(s)* a ser(em) criado(s) em resposta ao *issue*.

Para cada issue, deverá ser iniciado pelo responsável **ao menos 1** *Pull Request* (PR), onde as modificações serão efetivamente desenvolvidas.

É obrigatório que **cada integrante** do time também **aprove ao menos 2 PRs** de colegas.

Uma apresentação final do trabalho deve ser feita em sala de aula, no dia **08/05/2022**, em horário a ser agendado junto ao professor. Nessa apresentação, deverão ser executadas a versão inicial e a versão final do projeto, e a equipe deverá apresentar a proposta, a evolução alcançada e o histórico Git completo desde o *snapshot* inicial (todos os *branches*, inclusive/principalmente auxiliares). Devem ser explicadas as tarefas desenvolvidas de modo que os colegas de classe entendam o histórico Git do período.

Cr terios de avalia  o

A nota final atribu da a cada membro ser  totalizada a partir dos fatores a seguir:

Fator	M�x.	Individ.	Exemplos de cr�terios aplicados
Proposta de Projeto	1	N�o	Entregue/corrigida no prazo, cumpre com os cr�terios estabelecidos
Evolu��o do projeto	4	N�o	<i>Issues</i> foram criados e resolvidos de modo satisfat�rio e em n�mero adequado, as atividades foram bem planejadas e distribu�das entre os membros, houve diferen�a significativa entre os <i>snapshots</i> inicial e final do projeto
Contribui��o no projeto	2	<u>Sim</u>	O participante utilizou as ferramentas de colabora��o/comunica��o (<i>issue</i> e PR) de modo adequado, produziu uma quantidade de c�digo condizente com a do restante do time, efetuou as opera��es de versionamento conforme visto em sala de aula, executou o n�mero especificado de opera��es de cria��o e aprova��o de PRs
Apresenta��o	3	N�o	A equipe apresentou um gr�fico do hist�rico de <i>commits</i> , exibiu os <i>snapshots</i> inicial e final em execu��o, comunicou satisfatoriamente a evolu��o apresentada no decorrer do projeto
Total	10		

Vale ressaltar que o fator “Contribui  o no projeto” ser  objeto de avalia  o individual, ou seja, a nota poder  diferir de integrante para integrante.