

**ESTÁCIO
NOVA IGUAÇU**

Desenvolvendo a Web com Propósito: Abordagens Didáticas para o Ensino Eficiente

Tayná Araujo Lomboni da Silva – 202403858037

Victória Martins Nascimento – 202402441256

Angélica Freitas dos Santos Mello – 202408224801

Jander Pereira

**2024
Nova Iguaçu – Rio de Janeiro**

1. DIAGNÓSTICO E TEORIZAÇÃO

1.1. Identificação das partes interessadas e parceiros

O projeto de desenvolvimento de uma página de Lista de Tarefas (To-do List) visa proporcionar uma plataforma prática para estudantes e desenvolvedores iniciantes aprimorarem suas habilidades em JavaScript. Os participantes deste projeto são majoritariamente estudantes de tecnologia e desenvolvedores iniciantes, os participantes deste projeto são majoritariamente estudantes de tecnologia e desenvolvedores iniciantes. A faixa etária dos participantes varia entre 18 e 30 anos, abrangendo jovens adultos que estão em fase de formação acadêmica ou início de carreira. Esta faixa etária é ideal para o desenvolvimento de habilidades técnicas e para a inserção no mercado de trabalho, buscando promover a diversidade e inclusão, com uma participação estimada de aproximadamente 60% de homens e 40% de mulheres. Esta distribuição de gênero reflete o esforço contínuo para equilibrar a representação de gênero na área de tecnologia.

1.2. Problemática e/ou problemas identificados

A problemática central que motivou a elaboração deste projeto de extensão é a **falta de habilidades práticas em programação web** entre estudantes e desenvolvedores iniciantes. Durante encontros e conversas com a comunidade acadêmica e tecnológica, foi identificado que muitos estudantes possuem conhecimento teórico em HTML, CSS e JavaScript, mas carecem de oportunidades para aplicar esses conhecimentos em projetos práticos. A funcionalidade de adicionar e remover itens de uma lista é uma tarefa comum em muitos projetos web. Ao dominar essa habilidade, os participantes estarão melhores preparados para enfrentar desafios mais complexos no futuro. Sendo assim, este projeto visa proporcionar uma experiência de aprendizado prática e enriquecedora para todos os envolvidos.

1.3. Justificativa

A problemática identificada, que é a falta de habilidades práticas em programação web entre estudantes e desenvolvedores iniciantes, é altamente pertinente do ponto de vista acadêmico. A aprendizagem baseada em projetos (PBL – Project-Based Learning) é uma metodologia educacional que enfatiza a produção e aplicação de conhecimentos para

resolver demandas reais. Este projeto de extensão, focado no desenvolvimento de uma página de Lista de Tarefas, exemplifica perfeitamente essa abordagem.

Relação com o Curso: O projeto está diretamente relacionado aos cursos de Tecnologia da Informação, Ciências da Computação e áreas afins. Os objetivos de formação desses cursos incluem:

Objetivo: Capacitar os estudantes a desenvolverem soluções web utilizando HTML, CSS e JavaScript.

Aprendizagem: Através do projeto, os estudantes aplicam conceitos teóricos em um contexto prático, desenvolvendo uma página funcional que permite adicionar e remover itens de uma lista de tarefas.

Resolução de Problemas Reais:

Objetivo: Preparar os estudantes para identificar e resolver problemas reais utilizando tecnologias da informação.

Aprendizagem: O projeto aborda uma necessidade prática comum no desenvolvimento web, permitindo que os estudantes pratiquem a manipulação do DOM com JavaScript, uma habilidade essencial no mercado de trabalho.

Trabalho em Equipe e Colaboração:

Objetivo: Promover habilidades de trabalho em equipe e colaboração entre os estudantes.

Aprendizagem: O projeto incentiva a colaboração entre os participantes, que trabalham juntos para desenvolver, testar e implementar a página de Lista de Tarefas.

1.4. Objetivos/resultados/efeitos a serem alcançados (em relação ao problema identificado e sob a perspectiva dos públicos envolvidos)

Desenvolver uma página web funcional que permita aos usuários adicionar e remover itens de uma lista de tarefas, utilizando HTML, CSS e JavaScript, com foco na manipulação do DOM.

Instrumentos de Avaliação: Testes de usabilidade, feedback dos usuários, e análise de funcionalidade através de relatórios de bugs e melhorias.

Capacitar os participantes do projeto com habilidades práticas em programação web, promovendo a aplicação de conhecimentos teóricos em um contexto real.

Instrumentos de Avaliação: Questionários de autoavaliação, avaliações práticas durante workshops, e acompanhamento do progresso individual através de portfólios de projetos.

Promover a colaboração e o trabalho em equipe entre os participantes, incentivando a troca de conhecimentos e experiências.

Instrumentos de Avaliação: Observação direta durante as atividades em grupo, feedback dos participantes sobre a dinâmica de equipe, e relatórios de atividades colaborativas.

1.5. Referencial teórico (subsídio teórico para propositura de ações da extensão)

A criação de um website em formato de to-do list é um projeto que atende à crescente demanda por ferramentas de organização pessoal e produtividade. No cenário contemporâneo, a administração eficiente do tempo e das tarefas diárias é essencial, não apenas no ambiente profissional, mas também no contexto educacional e pessoal. Portanto, o desenvolvimento de uma ferramenta digital acessível, prática e eficiente se faz necessário para auxiliar indivíduos a gerenciar suas responsabilidades de maneira organizada. O embasamento teórico para esse projeto deve incluir autores que abordam a produtividade, a experiência do usuário e o design de sistemas digitais.

David Allen, autor de “A Arte de Fazer Acontecer”, é uma das principais referências no campo da gestão de produtividade pessoal. Sua metodologia de produtividade GTD (Getting Things Done) serve como base teórica para o desenvolvimento de ferramentas como uma to-do list. Allen enfatiza a importância de externalizar tarefas e compromissos da mente para um sistema confiável, liberando assim espaço mental para focar no que realmente importa. O sistema GTD propõe a criação de listas organizadas por contexto, prioridade e prazos, facilitando a administração das atividades. Ao implementar

essas ideias em um website de to-do list, busca-se estruturar uma ferramenta que permita aos usuários capturar, processar e organizar suas tarefas de maneira eficiente.

Outro referencial teórico importante é Jakob Nielsen, pioneiro na área de usabilidade e design de interface com o usuário. Nielsen, em sua obra "Usability Engineering", apresenta princípios fundamentais para o desenvolvimento de interfaces digitais que são fáceis de usar e intuitivas. Entre esses princípios, destacam-se a simplicidade, a consistência e o feedback imediato ao usuário. No contexto de um website de to-do list, a aplicação desses conceitos garante que a ferramenta seja acessível para diversos perfis de usuários, desde iniciantes até aqueles mais familiarizados com tecnologias digitais. Uma interface bem projetada diminui o esforço cognitivo dos usuários e aumenta a eficácia da ferramenta.

Por fim, Don Norman, em seu livro "O Design do Dia A dia", contribui com a compreensão da importância do design centrado no usuário. Norman argumenta que o sucesso de uma ferramenta digital depende diretamente de sua capacidade de se adaptar às necessidades e comportamentos dos usuários. Ele introduz o conceito de "design invisível", onde o objetivo é criar sistemas que funcionem de forma intuitiva, sem que o usuário precise pensar no processo. No desenvolvimento do website de to-do list, o design centrado no usuário é fundamental para garantir que a ferramenta seja uma extensão natural das atividades diárias, promovendo a sensação de controle e fluidez nas ações de organização.

2. PLANEJAMENTO E DESENVOLVIMENTO DO PROJETO

2.1. Plano de trabalho (usando ferramenta acordada com o docente)

O objetivo central é desenvolver um website funcional de to-do-list que permita aos usuários organizar suas tarefas diárias de forma simples e eficiente. O website deve apresentar uma interface amigável, ser acessível em diferentes dispositivos e incluir funcionalidades como adição, edição, categorização e exclusão de tarefas, além de um sistema de notificação de prazos.

Cronograma

Fase Ação Prazo

Fase 1: Planejamento Definição de requisitos

Fase 2: Planejamento	Criação de wireframes e layout
Fase 3: Planejamento	Desenvolvimento do front-end do website
Fase 4: Planejamento	Implementação do back-end
Fase 5: Planejamento	Testes de funcionalidade e compatibilidade
Fase 6: Planejamento	Correção de bugs, ajustes e lançamento.

Para atingir os objetivos do projeto, é essencial elaborar um plano de trabalho detalhado que contemple todas as ações a serem executadas, com um cronograma definido, responsáveis por tarefa, recursos necessários e formas de acompanhamento dos resultados.

Plano de Trabalho:

O plano pode ser formulado de forma digital, utilizando ferramentas de gerenciamento de projetos, ou de maneira física, como cartolinas, quadro branco e murais em sala de aula, promovendo a interação e o acompanhamento coletivo.

Ações a serem Executadas:

1. Planejamento Inicial:

Descrição: Fizemos pesquisas para entrar em um conhecimento que pudesse beneficiar como um todo.

Responsável: Coordenador do projeto (Tayná Lomboni)

Prazo: 2 semanas.

2. Pesquisa e Desenvolvimento:

Descrição: Habilidades e práticas são mais eficazes do que “apenas” teoria.

Responsável: Victória Martins

Prazo: 4 semanas.

3. Implementação:

Descrição: Colocação em prática do roteiro desenvolvido, monitorando sua execução.

Responsável: Equipe de implementação (Angélica Mello)

Prazo: 6 semanas.

Cronograma:

Planejamento Inicial: Semanas 1-2

Pesquisa e Desenvolvimento: Semanas 3-6

Revisão e Ajustes: Semanas 7-8

Implementação: Semanas 9-14

Recursos Necessários:

Materiais: Equipamentos de informática, materiais de escritório, acesso a softwares específicos.

Institucionais: Espaços físicos como salas de aula, laboratórios, bibliotecas.

Humanos: Professores como coordenadores e supervisores, alunos como participantes e estagiários.

Acompanhamento dos Resultados:

Conformidade com o Cronograma:

Indicador: Percentual de tarefas concluídas dentro do prazo.

Qualidade do Conteúdo:

A elaboração de um plano de trabalho claro e detalhado é fundamental para garantir a execução eficiente do projeto de extensão, promovendo um acompanhamento contínuo e permitindo ajustes necessários ao longo do processo.

2.2. Descrição da forma de envolvimento do público participante na formulação do projeto, seu desenvolvimento e avaliação, bem como as estratégias pelo grupo para mobilizá-los.

Desenvolvimento do Projeto: Interação e Construção Conjunta

Com o levantamento inicial das necessidades, a etapa de desenvolvimento do website foi guiada pela interação contínua entre o público acadêmico e a comunidade local. Cada etapa de desenvolvimento foi realizada de forma transparente, onde o grupo acadêmico apresentou protótipos e recebeu feedback constante dos usuários potenciais. Essa colaboração garantiu que o projeto não perdesse sua relevância e utilidade prática.

Durante o desenvolvimento, algumas estratégias de mobilização foram adotadas para manter o público engajado. Isso incluiu a criação de um grupo de discussão online, onde os participantes podiam sugerir melhorias e discutir funcionalidades específicas. Além disso, o uso de formulários de avaliação durante o processo ajudou a coletar feedback específico em cada fase do projeto, desde a interface de usuário até questões técnicas.

As funcionalidades como listas personalizáveis, modos de visualização diferentes (diário, semanal e mensal) e lembretes automáticos foram incorporadas após esses diálogos, refletindo o resultado da colaboração direta entre a equipe de desenvolvimento e os usuários finais.

2.3. Grupo de trabalho (descrição da responsabilidade de cada membro)

Gerente de Projeto (Tayná Lomboni): O gerente de projeto atua como o principal responsável pela coordenação geral do projeto. Ele garante que todas as etapas do desenvolvimento do website sejam cumpridas dentro dos prazos e que os objetivos propostos sejam atingidos. Entre suas atividades principais estão o agendamento e a condução das reuniões, o acompanhamento das atividades dos demais membros e a comunicação direta com o público-alvo. Além disso, o gerente é responsável por gerenciar os recursos disponíveis, planejar cronogramas e solucionar eventuais conflitos dentro da equipe.

Designer de Interface e Experiência do Usuário - (Victória Martins) (UI/UX Designer): O designer UI/UX é o responsável por criar a identidade visual do website e pela experiência do usuário (UX). Ele desenvolve o layout, cores, tipografias e os ícones, além de criar o fluxo de navegação que facilita o uso do website. Sua função é projetar uma experiência agradável e eficiente, conduzindo testes de usabilidade e coletando feedback para realizar

ajustes na interface. O objetivo do designer é garantir que o usuário consiga interagir com a plataforma de forma intuitiva, sem dificuldades para criar, editar e acompanhar suas tarefas.

Responsável pela Comunicação e Feedback - (Angélica Mello) Esse profissional tem como função estabelecer e manter uma comunicação ativa com o público-alvo, coletando feedbacks ao longo do desenvolvimento do projeto. Ele organiza pesquisas, conduz entrevistas e facilita grupos focais para obter insights sobre as funcionalidades desejadas pelos usuários e as melhorias que podem ser feitas no website. Além disso, ele mantém todos os stakeholders informados sobre o andamento do projeto e sobre as implementações realizadas com base nas sugestões recebidas.

2.4. Metas, critérios ou indicadores de avaliação do projeto

A primeira etapa envolve o Planejamento Inicial, onde são definidos os objetivos específicos, os recursos necessários e o cronograma geral do projeto. Nesta fase, também se estabelece a equipe responsável por tarefa, assegurando clareza nas atribuições e responsabilidades.

Na segunda etapa, focamos na Pesquisa e Desenvolvimento. Aqui, coletamos dados relevantes e desenvolvemos o conteúdo do roteiro de extensão. Esta fase é essencial para assegurar a qualidade e a consistência do material, exigindo uma análise aprofundada e a integração das melhores práticas e conhecimentos disponíveis.

A terceira etapa é a Revisão e Ajustes. Após o desenvolvimento inicial, o conteúdo precisa ser revisado e ajustado conforme necessário. Esse processo envolve feedback de revisores e stakeholders, garantindo que o roteiro atenda aos padrões esperados de qualidade e relevância.

Finalmente, na etapa de Implementação, colocamos o roteiro em prática. Este é o momento de monitorar a execução do projeto, avaliar o desempenho e realizar as correções necessárias em tempo real. A implementação eficaz requer um acompanhamento contínuo e uma comunicação aberta entre todos os membros da equipe.

Para avaliar a efetividade do projeto, utilizamos critérios como a conformidade com o cronograma, a qualidade das entregas e a satisfação dos stakeholders. Indicadores como a taxa de conclusão das tarefas, o tempo despendido em cada etapa e o feedback dos participantes são fundamentais para monitorar o progresso e ajustar o planejamento

conforme necessário. Assim, garantimos a entrega final com qualidade e dentro do prazo estipulado, atingindo os objetivos previstos na seção 1.4.

2.5. Recursos previstos

Para o desenvolvimento do projeto de extensão, é essencial considerar a previsão de recursos materiais, institucionais e humanos, buscando minimizar ao máximo os custos financeiros devido à ausência de previsão orçamentária específica nas Instituições de Ensino Superior (IES).

Recursos Materiais:

Os recursos materiais necessários englobam itens como equipamentos de informática, materiais de escritório e possíveis ferramentas específicas relacionadas ao projeto. A obtenção desses itens pode ser facilitada através de parcerias com empresas ou instituições locais que possam fornecer doações ou empréstimos. A reutilização de materiais existentes na instituição também é uma estratégia viável para reduzir custos.

Recursos Institucionais:

O apoio institucional é crucial para o sucesso do projeto. Isso inclui o uso de espaços físicos da instituição, como salas de aula, laboratórios e bibliotecas, sem a necessidade de novos investimentos financeiros. A disponibilização de recursos de tecnologia da informação, como acesso à internet e software específico, também pode ser proporcionada pela IES sem a geração de novos custos.

Recursos Humanos:

O envolvimento de professores, alunos e colaboradores da instituição é fundamental. Professores podem atuar como coordenadores e supervisores do projeto, enquanto alunos podem participar como estagiários ou voluntários, ganhando experiência prática. A valorização do trabalho voluntário, com possíveis certificados de participação, pode motivar a adesão de mais pessoas, reduzindo a necessidade de contratações externas.

Gestão de Custos:

Qualquer indicação de gastos financeiros deve ser acompanhada da definição clara da fonte desses recursos. Isso pode incluir o orçamento interno da instituição, parcerias com outras instituições ou empresas, e possíveis captações de recursos através de editais e

financiamentos específicos para projetos de extensão. A busca por estratégias que minimizem custos, como o uso de recursos já disponíveis e o estabelecimento de parcerias, é essencial para a viabilidade do projeto.

Com essas medidas, é possível desenvolver um roteiro de extensão eficaz e sustentável, alinhado com as possibilidades financeiras e estruturais da instituição.

2.6. Detalhamento técnico do projeto

Para o desenvolvimento do projeto de extensão, especialmente na criação de uma solução de Tecnologia da Informação (TI) conforme as etapas definidas no item 14 – Procedimentos de Ensino-Aprendizagem do Plano de Ensino, etapa 4, é necessário seguir uma abordagem estruturada e detalhada.

Planejamento Inicial:

O primeiro passo é a definição clara dos objetivos da solução de TI, incluindo a identificação das necessidades dos usuários e a especificação dos requisitos funcionais e não funcionais. Esta etapa envolve reuniões com os stakeholders para alinhamento das expectativas e a elaboração de um cronograma que distribua as tarefas ao longo do tempo.

Desenvolvimento:

Com o planejamento consolidado, inicia-se a fase de desenvolvimento. Aqui, a equipe de TI deve começar a codificar a solução conforme os requisitos definidos, utilizando metodologias ágeis para permitir a flexibilidade e ajustes contínuos. Testes preliminares são conduzidos durante o desenvolvimento para garantir que o produto esteja se alinhando às expectativas.

Implementação:

Após o desenvolvimento, a solução de TI é implementada em um ambiente de teste para identificar e corrigir possíveis falhas. A seguir, a implementação é expandida para o ambiente de produção. Nessa fase, é crucial um acompanhamento rigoroso para garantir que a solução funcione conforme o esperado em um cenário real.

Treinamento e Suporte:

Uma vez que a solução de TI está operacional, é fundamental oferecer treinamento aos usuários finais, garantindo que todos estejam capacitados para utilizar as novas

ferramentas de forma eficiente. Além disso, deve ser disponibilizado suporte contínuo para resolver eventuais problemas e aprimorar a solução com base no feedback dos usuários.

Avaliação:

Finalmente, a efetividade da solução é avaliada através de indicadores como a taxa de adoção pelos usuários, a redução de erros e o ganho de eficiência nas atividades propostas. O feedback coletado nesta etapa é essencial para ajustes e melhorias contínuas, assegurando que a solução de TI atenda plenamente às necessidades dos usuários e alcance os objetivos estabelecidos.

Através dessas etapas, é possível desenvolver uma solução de TI eficaz e alinhada com os objetivos do projeto de extensão, garantindo um resultado bem-sucedido e de qualidade.

3. ENCERRAMENTO DO PROJETO

3.1. Relato Coletivo:

Na fase inicial, a equipe se reuniu para discutir as funcionalidades que a to-do list deveria ter. Decidimos incluir:

1. Adicionar Tarefas: Permitir que os usuários possam criar entradas na lista.
2. Marcar como Completas: Oferecer a possibilidade de marcar tarefas como concluídas.
3. Remover Tarefas: Incluir uma opção para deletar tarefas.
4. Filtro de Tarefas: Criar uma funcionalidade para filtrar tarefas com base em seu status (todas, completas, pendentes).
5. Persistência de Dados: Utilizar local storage ou uma API para armazenar as tarefas e garantir que elas permaneçam salvas após a atualização da página.

Desenvolvimento

Dividimos as responsabilidades entre os membros da equipe. Enquanto alguns se concentraram no design da interface (UI/UX), outros trabalharam na lógica de programação e no banco de dados.

Design: Criou um layout simples e intuitivo, focando na usabilidade. Após várias iterações de feedback, chegamos a um modelo que agradou a todos.

- Programação: Optamos por utilizar JavaScript junto com HTML e CSS. Implementamos a funcionalidade de adicionar e remover tarefas com event listeners e manipulação do DOM, e utilizamos o local storage para a persistência dos dados.

Desafios Encontrados

Durante o desenvolvimento, enfrentamos alguns desafios:

- Sincronização de Estado: A equipe teve que trabalhar na sincronização do estado da aplicação com o armazenamento local, garantindo que alterações feitas em várias abas do navegador fossem refletidas corretamente.
- Responsividade: Criar uma interface que funcionasse bem em diferentes dispositivos foi outro desafio. Precisamos aplicar conceitos de design responsivo que permitiram que a to-do list fosse acessível em celulares, tablets e desktops.
- Gerenciamento do Tempo: Com prazos apertados, tivemos que administrar nosso tempo de forma eficaz, priorizando as funcionalidades mais essenciais primeiros.
- Resultados

Após várias semanas de colaboração, lançamos nossa to-do list funcional. A equipe ficou muito satisfeita com o resultado final e com o processo de colaboração, que foi repleto de aprendizado e troca de conhecimentos.

Feedback dos Usuários

Após o lançamento, coletamos feedback dos usuários e fizemos alguns ajustes com base nas sugestões. Por exemplo, implementamos um sistema de notificações que alerta os usuários sobre tarefas pendentes e a possibilidade de categorizar as tarefas por importância.

Reflexões Finais

O desenvolvimento da to-do list nos proporcionou uma valiosa experiência de trabalho em equipe, aumentando nosso entendimento sobre desenvolvimento web. Aprendemos a importância da comunicação clara e da divisão de tarefas, além de apreciar a aplicabilidade do feedback contínuo. Além de alcançar nosso objetivo, a experiência nos uniu como grupo e fortaleceu nosso espírito colaborativo, preparando-nos para futuros projetos.

Essa to-do list não foi apenas uma ferramenta útil; tornou-se um símbolo do que podemos alcançar quando trabalhamos juntos.

```
Z:\home\dataflair\Downloads\projects\project_to_do_list.html - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
project_to_do_list.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <link rel="stylesheet" type="text/css" href="css/to do list.css">
5 </head>
6 <body>
7
8 <div class="header">
9 <h2 style="margin:5px">To Do List</h2>
10 <input type="text" id="myInput" placeholder="Add content here">
11 <span onclick="newElement()" class="addBtn">Add</span>
12 </div>
13
14 <ul id="myUL">
15 <li class="checked">Go through DataFlair's JavaScript tutorials</li>
16 <li>Practise the codes</li>
17 <li>Read a book on JavaScript</li>
18 <li class="checked">Organise my notes</li>
19 <li>Create JavaScript projects</li>
20 <li>Take quiz</li>
21 <li>Prepare for interview questions</li>
22 </ul>
Hyper Text Markup Language file length: 854 lines: 29 Ln: 1 Col: 1 Sel: 0 | 0 Unix (LF) UTF-8 INS
```

```
Z:\home\dataflair\Downloads\projects\css\to do list.css - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
to do list.css
1 body {
2 margin: 10px auto;
3 min-width: 250px;
4 max-width: 50%;
5 background-color: silver;
6 }
7 * {
8 box-sizing: border-box;
9 }
10
11 ul {
12 margin: 0;
13 padding: 0;
14 }
15 ul li {
16 cursor: pointer;
17 position: relative;
18 padding: 12px 8px 12px 40px;
19 list-style-type: none;
20 background: #eee;
21 font-size: 18px;
22 transition: 0.2s;
Cascade Style Sheets File length: 1,902 lines: 116 Ln: 1 Col: 1 Sel: 0 | 0 Unix (LF) UTF-8 INS
```



```

Z:\home\dataflair\Downloads\projects\js\to do list.js - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
to do list.js
1 // Create a "close" button, append it to each list item
2 var myNodeList = document.getElementsByTagName("LI");
3 var i;
4 for (i = 0; i < myNodeList.length; i++) {
5     var span = document.createElement("SPAN");
6     var txt = document.createTextNode("\u00D7");
7     span.className = "close";
8     span.appendChild(txt);
9     myNodeList[i].appendChild(span);
10 }
11
12 // Click on a close button: hides the current list item
13 var close = document.getElementsByClassName("close");
14 var i;
15 for (i = 0; i < close.length; i++) {
16     close[i].onclick = function() {
17         var div = this.parentElement;
18         div.style.display = "none";
19     }
20 }
21
22 // Add a "checked" sign when clicking on a list item

```

JavaScript file | length: 1,731 | lines: 61 | Ln: 1 | Col: 1 | Sel: 0 | 0 | Unix (LF) | UTF-8 | INS