

StudySphere

Mariana Andrade

marianaandrade@ua.pt

103823

<https://marianaandrad.github.io/mei-tdw-project1/>

<https://github.com/MarianaAndrad/mei-tdw-project1>



Mestrado em Engenharia Informática

27 de outubro de 2024

0.1 Introdução

No âmbito da unidade Curricular Tecnologias e Desenvolvimento Web foi proposto o desenvolvimento de um mini-projeto, com o objetivo de aplicar os principais conceitos de introdução à programação para um ambiente web. Deste modo, foi desenvolvido o **StudySphere**, uma plataforma digital que tem como objetivo aumentar a produtividade e promover a entreaajuda entre os utilizadores. Esta plataforma foi desenvolvida com o objetivo de aplicar os conceitos aprendidos em contexto académico, bem como demonstrar alguns conceitos explorados em tempo extra-curricular de forma a completar a experiência académica.

O **StudySphere** utiliza a técnica de gestão de tempo *Pomodoro* ¹, criando um ambiente de trabalho partilhado onde os utilizadores comunicam possíveis dúvidas mantendo a motivação ao longo da sessão de estudo. Com um cronómetro integrado, a plataforma permite configurar o tempo de foco de forma personalizada, promovendo uma gestão eficaz de tempo e prevenindo o *burnout* ². Nos intervalos, os utilizadores podem participar em mini-jogos, adicionando um elemento lúdico à sua rotina e tornando o processo mais divertido.

Além disso, o **StudySphere** permite criar listas personalizadas de acordo com os objetivos do utilizador, de maneira a encorajar um estudo eficiente e metódico.

¹A técnica *Pomodoro* consiste em dividir o período de estudo em intervalos de concentração intensa, geralmente 25 minutos, intercalados por curtos períodos de descanso.

²*Burnout* é um problema comum entre estudantes caracterizado pela rápida desmotivação devido a uma sobrecarga de trabalho

0.2 Descrição dos Elementos Dinâmicos

O **StudySphere** apresenta um conjunto de elementos dinâmicos que alia a interatividade de *JavaScript* com as capacidades de representação visual do *CSS*.

0.2.1 Página Inicial (index.html)

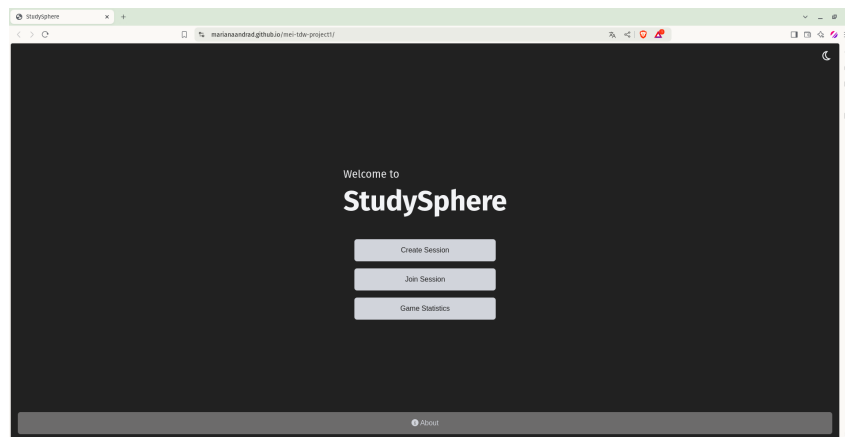


Figure 1: Interface Inicial

A página inicial tem como objetivo fornecer uma primeira conexão com o utilizador, e como tal, foi decidido manter a página simples, mas direta ao ponto. Nesta interface, torna-se possível criar uma sessão de estudo ou juntar-se a uma, visualizar estatísticas sobre sessões de estudo anteriores ou conhecer um pouco mais a ideia através do botão de *about*. Todos os elementos são dinâmicos, permitindo aceder às páginas sinalizadas com recurso a *JavaScript*.

0.2.2 Página de Criação da Sessão (create-session.html)

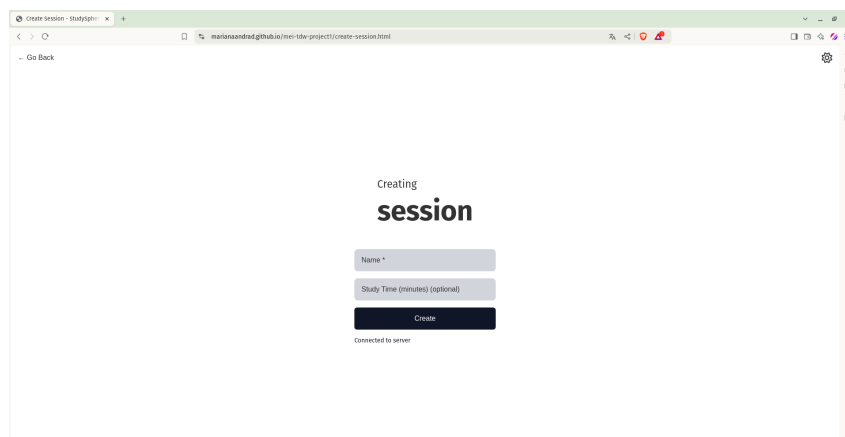


Figure 2: Interface para criar uma nova sessão

Nesta interface, os utilizadores têm a possibilidade de criar uma nova sessão de estudo personalizada, onde devem inserir o seu nome e, opcionalmente, definir o tempo de intervalo de concentração da sessão. Caso a duração não seja especificada, o tempo padrão será de 25 minutos. O sistema *P2P*³, gerido através do *PeerJS* usando **WebRTC**, é responsável por gerar um código único de sessão, que pode ser partilhado

³Comunicação Peer to Peer

com outros utilizadores, permitindo que estes participem na mesma sessão de estudo. Após a criação da sessão, o cronómetro *Pomodoro* é iniciado, gerindo automaticamente os períodos de foco e descanso conforme a configuração escolhida.

0.2.3 Página de Entrar numa Sessão (join-session.html)

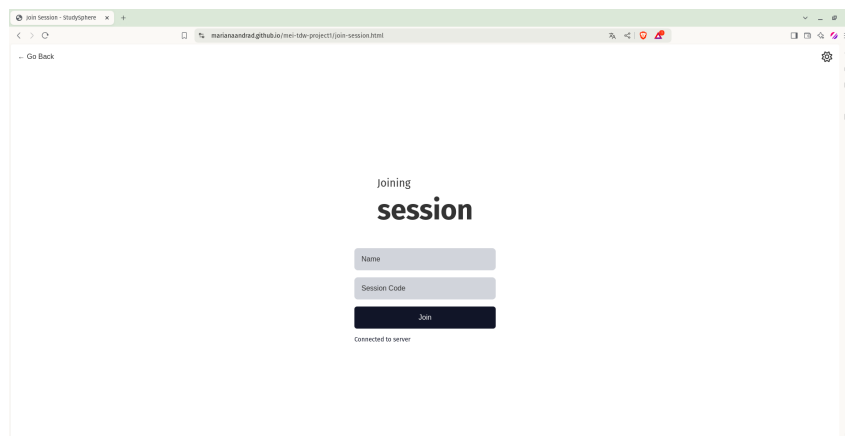


Figure 3: Interface para o utilizador entrar numa sessão

Na figura 3, os utilizadores podem entrar numa sessão existente, colocando um nome e o código da sessão fornecido por outro participante. Após a validação dos dados executada em *JavaScript*, o utilizador é redirecionado para a sessão. Todo processo de comunicação é gerido em tempo real através da framework *PeerJS*, possibilitando a conexão direta entre utilizadores numa rede P2P.

0.2.4 Página da Sessão (session.html)

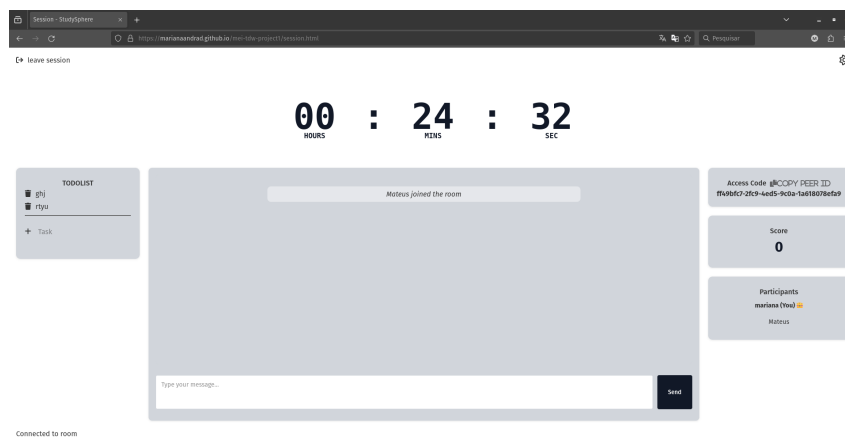


Figure 4: Interface da sessão de trabalho

Ao entrar numa sessão, o utilizador tem acesso a diversas funcionalidades.

- **Lista de tarefas Interativa**

Os utilizadores podem adicionar ou eliminar tarefas pessoais. Esta funcionalidade é implementada com recurso a uma API, que armazena e recupera as tarefas de forma eficiente, garantindo a persistência dos dados entre sessões.

- **Cronómetro**

O cronómetro controla o tempo de trabalho, permitindo que todos os utilizadores sigam o mesmo ciclo de foco e pausa. O temporizador é sincronizado entre os participantes, garantindo que todos permanecem na mesma fase da sessão.

- **Chat**

A plataforma inclui um Chat integrado que permite a comunicação entre os participantes da sessão. As mensagens são enviadas e recebidas em tempo real através da mesh P2P.

- **Mini-jogo**

Quando o tempo do cronómetro termina, uma modal com um mini-jogo de trivial é apresentada, como ilustrado na figura 5. Esta funcionalidade consome recursos de uma API externa. Após todos os jogadores concluírem o jogo, o cronómetro é reiniciado automaticamente para todos os participantes.

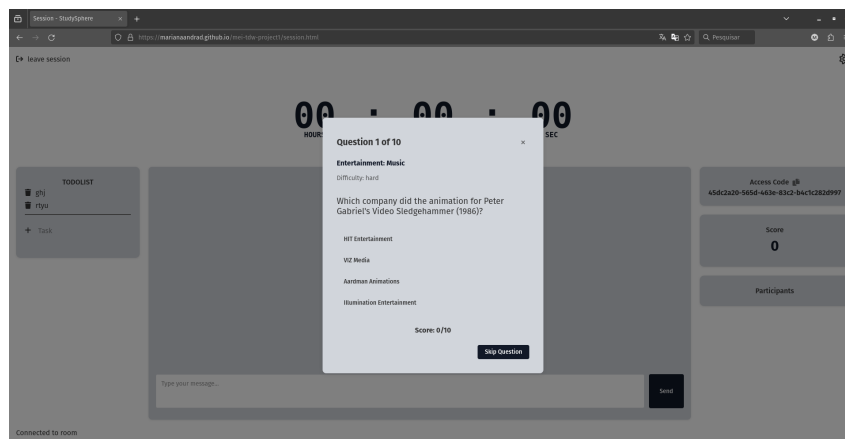


Figure 5: Interface Jogo no tempo de Intervalo

0.2.5 Página de Estatísticas (stats.html)

Nesta página, os utilizadores acompanham o desempenho nos mini-jogos dos intervalos de estudo. As estatísticas, calculadas com *JavaScript*, mostram o número total de jogos, pontuação média, mais alta e mais baixa. A tabela de histórico, gerada a partir de dados guardados em *localStorage*, mostra os resultados obtidos, incluindo data e pontuação.

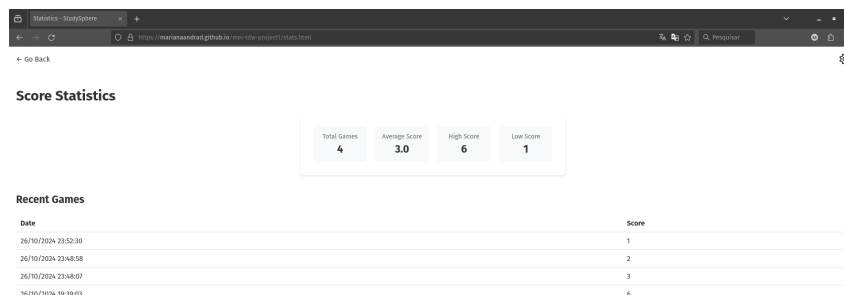


Figure 6: Pagina de Estatísticas dos mini-jogos

0.2.6 Troca de Tema

O **StudySphere** permite aos utilizadores alternar entre os temas claro e escuro conforme suas preferências. Esta funcionalidade é implementada em *JavaScript*, que altera as classes *CSS* para modificar o estilo da página. A escolha do utilizador é armazenada no *LocalStorage*, garantindo que a preferência é mantida mesmo após recarregar a página ou sair da sessão.

0.3 Estratégia de Implementação

O projeto foi desenvolvido utilizando o trio de ferramentas de eleição para a Web, nomeadamente **HTML**, **CSS** e **JavaScript**, juntamente com a biblioteca PeerJS para comunicação P2P e Web Standards como WebRTC⁴ para comunicação em tempo real.

0.3.1 Estrutura do Projeto

A organização do projeto segue uma estrutura modular, dividida em pastas específicas:

- **Páginas HTML:** Cada funcionalidade principal corresponde a um ficheiro HTML dedicado
 - index.html - Página inicial
 - create-session.html - Criação de sessão
 - join-session.html - Entrada em sessão
 - session.html - Página principal da sessão
 - stats.html - Estatísticas
 - about.html - Página da Informação do StudySphere
- **/css:** Contém todos os ficheiros de estilo
 - styles.css - Estilos globais
 - theme.css - Definições de tema claro/escuro
- **/js:** Componentes dinâmicos separados por funcionalidade
 - chat.js - Gestão da comunicação P2P, sincronização, Integração com API externa⁵ para implementação do mini-jogo
 - todos.js - Integração com API externa⁶ e gestão de tarefas
 - theme.js - Gestão do tema da interface
 - script-stats.js - Gestão das estatísticas dos mini-jogos
 - script-join-session.js - Gestão da entrada dos utilizadores numa sessão
 - script-create-session.js - Gestão da criação de uma sessão

⁴<https://webrtc.org/>

⁵trivial questions API - <https://rapidapi.com/tethertechnologies095/api/trivia-questions-api>

⁶Criada usando <https://mockapi.io/>

0.3.2 Gestão de Estado e Persistência

A persistência de dados foi implementada considerando dois níveis:

- **SessionStorage:** Utilizado para dados temporários que devem persistir apenas durante a sessão atual, como:
 - Estado atual do temporizador
 - Informações da sessão atual
- **LocalStorage:** Empregue para dados que devem persistir entre sessões, como:
 - Preferências de tema (claro/escuro)
 - Histórico de pontuações dos mini-jogos
 - Configurações personalizadas do utilizador

0.3.3 Padrões de Desenvolvimento

O desenvolvimento seguiu princípios de modularidade e separação de responsabilidades:

- Cada componente JavaScript é responsável por uma única funcionalidade principal
- Utilização de eventos personalizados para comunicação entre componentes
- Implementação de padrões de design para gestão de estado e atualizações da interface

0.4 Principais Desafios Técnicos

Durante o desenvolvimento do **StudySphere**, foram encontrados diversos desafios técnicos significativos que exigiram soluções criativas e robustas. O maior desafio técnico encontrado foi a criação de um protocolo de comunicação que permitisse a comunicação eficiente entre estudantes. Este desafio complicou-se uma vez que WebRTC é um standard de comunicação entre pares de utilizadores. Por este motivo, foi decidido que o criador da sala serviria como intermediário entre todas as comunicações, implementando uma topologia em estrela onde todas as mensagens passam pelo host da sessão.

A otimização da performance da aplicação requereu um cuidado especial com a gestão de memória, especialmente em conexões P2P de longa duração. Foi necessário implementar estratégias de throttling e debouncing para minimizar o tráfego de rede.

Garantir uma experiência consistente entre diferentes navegadores e dispositivos também foi desafiador. A interface precisou ser adaptada para diferentes tamanhos de ecrã, e foram necessários fallbacks para funcionalidades não suportadas em navegadores mais antigos, especialmente considerando as diferentes implementações de WebRTC entre navegadores.

As soluções para estes desafios foram implementadas de forma iterativa, priorizando sempre a estabilidade e a experiência do utilizador. A arquitetura modular adotada permitiu isolar e resolver problemas específicos sem afetar outras partes do sistema, resultando numa aplicação robusta e funcional.

0.5 Conclusões

O desenvolvimento do **StudySphere** permitiu a consolidação dos conceitos abordados na unidade curricular, durante o modulo I. Além de aplicar os fundamentos da programação web, o projeto desafiou-nos a explorar tecnologias avançadas como o WebRTC e a comunicação P2P, ampliando o nosso conhecimento para além do currículo básico.

A plataforma, apesar de simples, trouxe alguns desafios técnicos, tanto em termos técnicos quanto de design. No futuro, funcionalidades adicionais podem ser introduzidas, como por exemplo uma listagem de jogo. Contudo, na sua versão atual, o **StudySphere** já cumpre o seu propósito de criar um ambiente produtivo, colaborativo e agradável para os estudantes, oferecendo uma base sólida para crescimento e inovação contínua.