



COORDENADORIA DE ENGENHARIA DE COMPUTAÇÃO

GUSTAVO SANCHES
RAFAEL ZABEU SPESSOTTO

PLATAFORMA GAMIFICADA DE TREINAMENTO EMPRESARIAL

Sorocaba/SP

2019

**Gustavo Sanches
Rafael Zabeu Spessotto**

PLATAFORMA GAMIFICADA DE TREINAMENTO EMPRESARIAL

**Sorocaba/SP
2019**

AGRADECIMENTOS

RESUMO

O objetivo do projeto é desenvolver uma aplicação de aprendizado gamificada para empresas utilizarem como reforço e incentivo para aprendizado de tópicos.

A aplicação permitiria a empresa criar questões sobre um assunto que considere interessante e acompanhar quantas pessoas acertaram e erraram cada questão, podendo assim verificar quanto esse assunto é dominado pelos participantes e acompanhar o progresso do aprendizado em geral.

Sendo uma plataforma gamificada os usuários são incentivados a jogar para subir no ranking geral, e respondendo as perguntas vai gradualmente aprendendo e reforçando os conhecimentos desejados.

A aplicação teria um servidor para guardar as informações e coordenar os jogos, clientes em PC, Android e IOS onde o jogo será jogado e uma aplicação web como dashboard administrativo.

Palavras-chave: Gamificação. Indústria. Unity3D. Lúdico.

ABSTRACT

O objetivo do projeto é desenvolver uma aplicação de aprendizado gamificada para empresas utilizarem como reforço e incentivo para aprendizado de tópicos.

A aplicação permitiria a empresa criar questões sobre um assunto que considere interessante e acompanhar quantas pessoas acertaram e erraram cada questão, podendo assim verificar quanto esse assunto é dominado pelos participantes e acompanhar o progresso do aprendizado em geral.

Sendo uma plataforma gamificada os usuários são incentivados a jogar para subir no ranking geral, e respondendo as perguntas vai gradualmente aprendendo e reforçando os conhecimentos desejados.

A aplicação teria um servidor para guardar as informações e coordenar os jogos, clientes em PC, Android e IOS onde o jogo será jogado e uma aplicação web como dashboard administrativo.

Keywords: Gamification. Corporative. Unity3D. Ludic.

LISTA DE ABREVIATURAS E SIGLAS

SUMÁRIO

1	INTRODUÇÃO	9
2	GAMIFICAÇÃO	10
2.1	Gamificação do aprendizado	10
2.2	Aplicações empresariais	10
3	DESENVOLVIMENTO DO PROJETO	11
3.1	Estrutura	11
3.2	REST API	11
3.2.1	Usuários	11
3.2.2	Controle de Acesso	12
3.2.3	Questões	12
3.2.3.1	Respostas	12
3.2.3.2	Escolhendo Questões	12
3.2.4	Partidas	13
3.2.5	Rankings	13
3.2.5.1	Reiniciando Rankings	14
3.3	Unity3D	14
3.3.1	Telas	14
3.3.1.1	Base	14
3.3.1.2	Login	15
3.3.1.3	Cadastro	15
3.3.1.4	Lobby	15
3.3.1.5	Ranking	15
3.3.1.6	Perguntas	16
3.3.1.7	Resultado	16
3.3.1.8	Carregando	16
3.3.1.9	Erros	16
3.3.2	Scriptable Objects Architecture	17
3.3.2.1	Scriptable Objects	17
3.3.2.2	Game Events	17
3.3.3	Acesso à API	18
3.4	React	18
3.4.1	Login	18
3.4.2	Questões	18
3.4.2.1	Detalhe da Questão	18

3.4.2.2	Relatório das Questões	18
3.4.3	Configurações	19
3.4.4	Rankings	19
3.4.5	Usuários	19
4	RESULTADOS	20
5	CONCLUSÃO	21
	REFERÊNCIAS	22
	APÊNDICE A – TELAS DA APLICAÇÃO	23

1 INTRODUÇÃO

2 GAMIFICAÇÃO

Este capítulo discutirá o que é gamificação e como ela pode ser utilizada no aprendizado. Também discutirá como a gamificação pode ser aplicada em um ambiente empresarial.

2.1 GAMIFICAÇÃO DO APRENDIZADO

Este capítulo apresentará como a gamificação pode ser aplicada ao aprendizado e efeito que ela tem no engajamento.

2.2 APLICAÇÕES EMPRESARIAIS

Este capítulo apresentará como a gamificação pode ser utilizada em um ambiente empresarial e dando embasamento teórico de como o projeto poderia ajudar uma empresa.

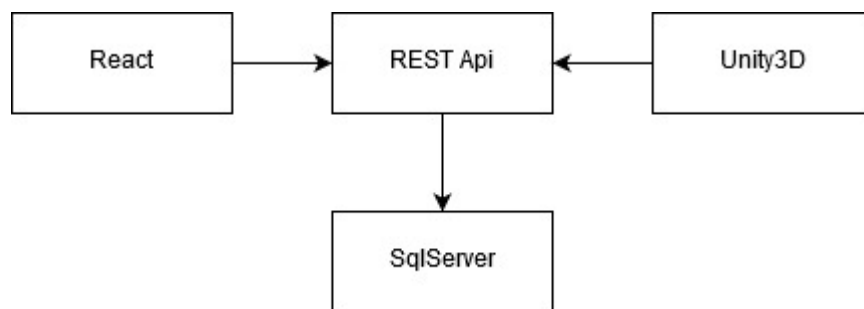
3 DESENVOLVIMENTO DO PROJETO

As ferramentas escolhidas para o desenvolvimento do projeto foram a Unity3D para a criação da aplicação, ASP.NET Core para a REST API e React para o dashboard administrativo.

3.1 ESTRUTURA

A Figura 3.1 demonstra o fluxo de dados entre os componentes do projeto, os clientes (Web ou Unity) fazem requisições para uma API que busca ou guarda os dados em um banco de dados relacional (SQL Server) e retorna o resultado da operação.

Figura 3.1 – Fluxo De Dados



Fonte: própria

3.2 REST API

A API serve como centro do projeto. Ela guarda todas as questões, usuários, opções e os rankings. Ela decide também quais e quantas questões um jogador irá receber quando uma nova partida for requisitada. Os clientes e a API se comunicam utilizando o protocolo Http(FIELDING, 2014) através do protocolo rest REST(?), enviando e recebendo JSONs.

A API é implementado em C# utilizando o ASP NET Core 2.2 e Entity Framework Core para acesso ao banco de dados.

3.2.1 Usuários

Para poder interagir com a maioria das funções da API é necessário que usuário crie uma conta e logue com ela. Cada usuário possui Username único que o identifica e uma lista de cargos que determinam quais funcionalidades da API o usuário tem

acesso. A senha do usuário é passada pelo algoritmo HMAC-SHA521 (KRAWCZYK M. BELLARE, 1997) e somente o hash resultante e o salt utilizados são salvos no banco.

Um usuário pode ser tanto um jogador como um administrador, podendo ser ambos dependendo dos cargos dados ao usuário.

A função da API para criação de usuários é a aberta, por tanto qualquer um pode se registrar, porém usuários criados assim só terão acesso às funções de jogador. Um administrador pode depois dar a um usuário acesso às funções de administração.

3.2.2 Controle de Acesso

Cada função da API restringe seu acesso dependendo dos cargos que ela requer de um usuário acessado. Existem 2 cargos no sistema “admin” e “user” que dão acesso às funções de administração e às funções do jogo respectivamente.

Algumas funções não requerem um usuário logado, como a função de cadastro e login, e outras só requerem que o usuário esteja logado, sem se importar com quais cargos eles possuem.

3.2.3 Questões

Cada questão possui um enunciado, uma flag indicando se essa questão está ativa e deveria ser enviada aos jogadores e um número aleatório utilizando para escolha das questões para um jogador. Opcionalmente uma questão pode ter uma imagem e varios Temas e Categorias.

Cada questão deve ter exatamente 4 respostas com somente uma sendo marcada como correta para poder ser marcada como ativa.

3.2.3.1 Respostas

Cada resposta tem um texto, uma flag indicando se essa resposta é correta e opcionalmente uma imagem.

Uma resposta está sempre ligada a uma única questão e somente uma resposta ligada a cada questão pode ser marcada como correta.

3.2.3.2 Escolhendo Questões

Quando um jogador começa uma nova partida a API escolhe quais questões para enviar para o jogador. A escolha é feita aleatoriamente entre as questões ativas que o jogador ainda não tenha respondido.

Tabela 3.1 – Exemplo de "ou exclusivo"

Input		Output
A	B	
0	0	0
0	1	1
1	0	1
1	1	0
54	90	108
1620668800	4194220096	2573715904

Primeiramente as questões são filtradas para conterem somente questões ativas que o jogador não tenha respondido ainda. Depois um número aleatório é gerado e é feito um “ou exclusivo” (demonstrado na Tabela 3.1) entre este número e o número aleatório de cada questão e as questões são ordenadas de acordo com os números resultantes. Por fim a quantidade necessária de questões é pega a partir do começo da lista.

Caso depois do processo não existam questões suficientes para uma partida, a lista de questões que o jogador já respondeu é apagada e o processo é repetido excluindo as questões que já tenham sido escolhidas anteriormente e o número de questões faltando são escolhidas. Se mesmo assim não houverem questões suficientes a partida começa com questões faltando.

3.2.4 Partidas

Uma partida começa com o jogador requisitando o começo de uma nova partida para a API e recebendo as questões a serem respondidas.

Uma vez que o jogador tenha respondido a todas as questões ele manda as respostas escolhidas de volta para a API que calcula quantos pontos o jogador fez nesta partida e quais questões foram respondidas.

Os pontos são adicionados a cada ranking e o total de pontos conseguidos e a nova classificação nos rankings é retornado ao jogador.

3.2.5 Rankings

O jogo possui 3 rankings por padrão, um semanal, um mensal que são reiniciados a cada 7 dias e a cada 30 dias respectivamente e um perpétuo que nunca é reiniciado. Os rankings são públicos e atualizados cada vez que uma partida termina.

3.2.5.1 Reiniciando Rankings

Os resultados dos rankings reiniciados não são perdidos. Ao invés de deletar os dados de um ranking, um novo ranking com o mesmo nome e duração é criado e o antigo é desativado, fazendo com que para os usuários o ranking seja reiniciado, mas para os administradores os dados ainda são acessíveis.

O processo de reiniciar os rankings é feito por um serviço periódico que roda a cada 30 minutos, porém um administrador pode reiniciar um ranking manualmente se desejado.

3.3 UNITY3D

O desenvolvimento da aplicação do projeto foi feita utilizando a Engine de jogos Unity3D. As principais vantagens obtidas por escolher a Unity3D como ambiente de desenvolvimento foram: a facilidade de criar projetos para plataformas diferentes utilizando a mesma base de código e design visual, um sistema intuitivo e visual de criação de telas e a familiaridade do Laboratório de Inovação de Games e Apps.(LIGA) com projetos feitos com a engine para suporte posterior.

<TODO> Game Object Cena Hierarquia

3.3.1 Telas

O design da aplicação foi montada para utilizar apenas uma cena dentro da Unity3D, isso faz com que não tenha tempo perdido para carregar outras telas durante a execução da aplicação, deixando-a mais responsiva para o usuário.

Não foram utilizados muitos elementos nas telas da aplicação para que o cliente possa customizar de acordo com o visual de seu negócio, modificando a paleta de cores e a disposição de logomarcas dentro da aplicação.

No total a aplicação conta com sete telas completas e dois elementos de suporte.

3.3.1.1 Base

A tela base consiste em uma cor padrão que preenche todo a parte de fundo da aplicação que pode ser modificada pelo cliente ou substituída por uma logomarca, e uma barra localizada na parte superior que contém o nome do usuário ativo, sua pontuação geral e o horário atual.

Todas as telas seguintes ficarão localizadas à frente da tela de fundo e sempre embaixo da barra superior, sendo essa sempre visível ao usuário.

3.3.1.2 Login

A tela de login é a tela principal, enquanto não há interação com a aplicação e após um usuário finalizar sua sessão ela ficará exposta. A tela contém três elementos.

O elemento central consiste em dois campos para serem preenchidos, um com o usuário e o outro com senha, e um botão para realizar o processo de login. Quando o usuário interagir com esse botão, caso algum dos campos esteja vazio, uma mensagem de erro aparecerá com uma mensagem informando o usuário para ele preencher todos os campos. Caso todos os campos estejam preenchidos, uma chamada para a API é feita, caso os valores estejam corretos o usuário é direcionado para a tela de seleção de temas para começar a experiência. Se as informações de login não estejam certas, uma mensagem de erro informando que os valores estão errados será mostrado ao usuário.

Os outros dois elementos consistem e dois botões localizados nos cantos da tela.

O botão no canto direito superior leva o usuário para a tela de ranking, onde ele poderá verificar a colocação de todos os usuários sem a necessidade de fazer login com uma conta.

O botão no canto direito inferior leva o usuário para a tela de cadastro.

3.3.1.3 Cadastro

A tela de cadastro contém uma área com campos necessários para a criação de uma nova conta, um botão para criar a conta com as informações informadas e um botão para retornar para a tela de login.

3.3.1.4 Lobby

Após o usuário realizar o login ele será redirecionado para a tela de lobby onde ele poderá iniciar uma sessão de perguntas para verificar seu conhecimento em uma área de trabalho. O usuário também tem acesso ao ranking e a possibilidade de voltar à tela de login, removendo o vínculo de sua conta com a sessão atual.

3.3.1.5 Ranking

A tela de ranking contém uma lista mostrando os melhores colocados em cada ranking.

3.3.1.6 Perguntas

A tela de perguntas é separada em duas partes, a área que mostra a pergunta escolhida pelo sistema, e a área dedicada às respostas que o usuário pode escolher.

A parte separada para a pergunta possui espaço suficiente para conter uma imagem e um pequeno texto explicativo para melhor entendimento do usuário. Esta parte é adaptativa e pode mudar seu tamanho dependendo do tamanho e quantidade de elementos presentes.

As respostas podem estar dispostas de duas maneiras. A primeira é como uma lista vertical caso as respostas sejam compostas somente por texto, deixando um espaço maior no eixo horizontal para deixar a leitura mais simples. Na segunda disposição, cada resposta ocupa um canto da área. Essa disposição permite que as respostas tenham uma imagem e um espaço para uma pequena descrição.

O usuário precisa escolher uma resposta, que ficará marcada como selecionada, e em seguida, precisa interagir com o botão “Próximo” para enviar sua seleção para o servidor. Logo em seguida, a resposta correta será indicada para o usuário, tendo um feedback instantâneo sobre sua escolha.

Quando o usuário estiver na última pergunta de uma partida o botão “Próximo” muda para mostrar o texto “Terminar” e quando o usuário receber o feedback da resposta escolhida, ele será redirecionado para a tela de resultados.

3.3.1.7 Resultado

A tela de resultados mostra a performance do usuário após responder as perguntas de uma sessão, informando-o do resultado da resposta de cada pergunta, o número de acertos em relação ao total de perguntas e a pontuação geral obtida na sessão.

3.3.1.8 Carregando

A tela de carregando aparece toda vez que alguma operação que requer que o usuário espere uma resposta. Esta tela aparece na frente de todas as outras e impede que o usuário interaja com a aplicação até receber uma resposta.

3.3.1.9 Erros

A janela de erros aparece na frente de todas as outras telas informando o usuário que algo de errado aconteceu junto de um botão que fecha a janela.

3.3.2 Scriptable Objects Architecture

No desenvolvimento da aplicação foram aplicados os conceitos sobre a utilização de Scriptable Object como base da arquitetura geral. O conceito As transições e comunicação entre scripts são feitas por meio de eventos que persistem durante a execução da aplicação como Scriptable Objects.

3.3.2.1 Scriptable Objects

Um ScriptableObject é um contêiner de dados utilizado para salvar grandes quantias de dados, independente de instâncias de classes. Um dos principais casos de uso para ScriptableObjects é para reduzir a utilização de memória de um projeto por evitar a cópia de valores entre múltiplos objetos.(UNITY, 2019)

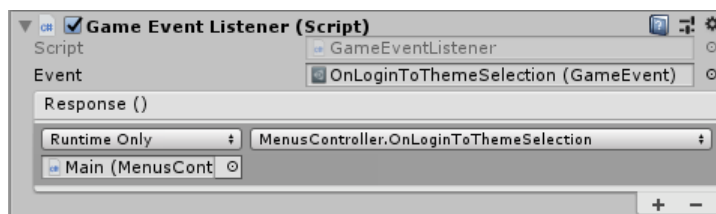
O Scriptable Obejct é composto de duas partes, o modelo e o objeto. <TODO>

3.3.2.2 Game Events

Game Events são scripts especiais derivados do Scriptable Object que podem enviar um sinal para vários Game Event Listener indicando uma mudança ou uma reação. Os Game Event Listeners são componentes que, quando dentro de um Game Object ativo em uma cena, ao receber um aviso de um Game Event identificado, responde chamando uma função especificada. Este sistema permite que Game Object respondam à mudanças de estado sem precisar monitorar uma variável constantemente.(HIPPLE, 2018)

A Figura mostra o componente Game Event Listener dentro de um Game Object, quando o evento OnLoginToThemeSelection é acionado, a função OnLoginToThemeSelection do script MenuController no Game Object Main. Isso permite que não seja necessária uma referência direta do quem acionou o evento e quem responde a ele, melhorando a organização do projeto ao fazer cada script controlar apenas seu próprio comportamento.

Figura 3.2 – Sistema de Game Event



Fonte: própria

3.3.3 Acesso à API

O acesso a API é feito através da biblioteca LigaFit. A LigaFit, inspirado pelo RetroFit mas implementado para Unity3D e levando em conta as características das plataformas normalmente utilizadas pelo LIGA, permite que a API seja descrita através de uma interface e gera implementação automaticamente.

3.4 REACT

As funções administrativas são disponibilizadas para o usuário através de um web site desenvolvido em React. O web site permite a um usuário administrador gerir as questões e usuários e obter informações sobre a taxa de acertos e erros das questões.

3.4.1 Login

A tela de login possui campos para o Username e a senha do usuário, além de uma opção para lembrar do login em visitas futuras.

Qualquer usuário pode logar no web site, porém somente os que tenham credenciais de administrador terão acesso a maioria das funções.

3.4.2 Questões

O web site mostra a lista de questões em ordem alfabética, junto com se a questão está ativa.

O usuário pode selecionar uma questão e ser redirecionado a tela de detalhes da questão.

3.4.2.1 Detalhe da Questão

A tela de detalhes de questão permite que o usuário edite o enunciado da questão, se ela está ativa, a imagem da questão e mostra e permite a edição das respostas da questão.

3.4.2.2 Relatório das Questões

Nesta tela o administrador pode ver um relatório sobre todas as questões, que mostra quantas vezes cada questão foi respondida e quantos erros, acertos e pulos uma questão teve. O relatório é pode ser configurado para mostrar informações de qualquer período de tempo que o administrador deseje.

3.4.3 Configurações

Esta tela permite que o administrador ajuste as configurações do jogo, podendo decidir quantas questões uma partida deve ter, quantos pontos cada questão vale em novas partidas.

3.4.4 Rankings

O ranking semanal, mensal e perpétuo são mostrados nesta tela. Como os rankings são públicos, essa tela não requerer que o usuário esteja logado para ser acessada.

3.4.5 Usuários

Os administradores podem visualizar todos os usuários da plataforma, podendo criar, editar e deletar usuários.

4 RESULTADOS

Apresenta a plataforma gamificada, explicando as funcionalidades e como elas são úteis para o aprendizado em uma empresa.

5 CONCLUSÃO

Lorem Ipsum

REFERÊNCIAS

FIELDING, J. R. R. ***Hypertext Transfer Protocol HTTP 1.1 Semantics and Content***. 2014. Acesso em: 25 set. 2019. Disponível em: <<https://tools.ietf.org/html/rfc7231#section-4>>.

HIPPLE, R. ***3 cool ways to architect your game with Scriptable Objects***. 2018. Acesso em: 24 set. 2019. Disponível em: <<https://unity3d.com/how-to/architect-with-scriptable-objects>>.

KRAWCZYK M. BELLARE, R. C. H. ***HMAC Keyed-Hashing for Message Authentication***. 1997. Acesso em: 25 set. 2019. Disponível em: <<https://tools.ietf.org/html/rfc2104>>.

UNITY. ***ScriptableObject***. 2019. Acesso em: 24 set. 2019. Disponível em: <<https://docs.unity3d.com/Manual/class-ScriptableObject.html>>.

APÊNDICE A – *TELAS DA APLICAÇÃO*