|  |  |
| --- | --- |
| Descrição: ssc.PNG | **UNIVERSIDADE DE SÃO PAULO**  **Instituto de Ciências Matemáticas e de Computação**  **Departamento de Sistemas de Computação** |

Relatório de Estágio

***Otávio Massola Sumi***

***Empresa****: Aptor Software*

##### Supervisor: Marcelo Petrucelli

|  |
| --- |
| Relatório de estágio realizado na empresa Aptor Software dentro do escopo da disciplina SSC0672 do Departamento de Sistemas de Computação do Instituto de Ciências Matemáticas e de Computação – ICMC-USP. |
|  |

**USP – São Carlos**

**29/05/2020**

## 1) Empresa onde o estágio foi realizado

A Aptor é uma empresa de consultoria e desenvolvimento de software voltada à criação e manutenção de soluções usáveis e eficazes. Localizada em São Carlos (Capital da tecnologia), posiciona-se estrategicamente em um centro de pesquisa e tecnologia principalmente originado por universidades bem reconhecidas.

Tem como foco principal a Usabilidade, em que o desenvolvimento dá prioridade às necessidades, às limitações, enfim, às características dos usuários do sistema. Deseja-se que as soluções desenvolvidas tenham o melhor aproveitamento possível com facilidade de uso.

Os diferenciais envolvidos na concepção de trabalho da Aptor estão na eficiência e qualidade de seus produtos e serviços, com o objetivo da inovação constante nas áreas computacionais para oferecer sempre a melhor solução aos seus clientes.

## 2) Setor da empresa onde o estágio foi realizado

O setor que o estágio foi realizado é a área de jogos educacionais da empresa, essa área é idealizada pelo CDMF e apoiada pela FAPESP e CNPq. O professor responsável até o momento deste relatório é o professor doutor Elson Longo, do departamento de materiais da UFSCAR.

A equipe tem como membros programadores e uma artista, no momento deste relatório a equipe contém 3 programadores e uma artista, cada projeto é de responsabilidade de cada programador em conjunto com a artista, apesar disso todos estão sempre a par dos projetos que são executados pelos outros programadores, permitindo assim discussão, resolução de problemas e tomada de decisões de projeto em conjunto com todos da equipe.

Os projetos dessa área podem ser conferidos na plataforma do “Ludo Educativo” [1].

## 3) Período em que o estágio foi realizado

O estágio teve início em novembro de 2018 até o presente momento.

## 4) Métodos, Técnicas e Tecnologias Utilizadas

### Método:

O método utilizado foi uma adaptação do SCRUM, fizemos uma reunião a cada semana, mostrando o que foi feito na semana que havia passado, como progressos no código, métodos de resolução interessante de problemas que podem ser comuns a outros projetos da área. Levantamos problemas que estávamos com dificuldade em resolver, dessa forma os outros desenvolvedores podiam contribuir com ideias e formas de resolução, todos (especialmente a artista) contribuíam quando era uma dúvida de design dos jogos ou UX. Depois dessas duas partes sempre levantávamos o que era pretendido para a próxima semana, a fim de conferir o progresso nos projetos, junto com anúncios importantes como datas de entrega ou novos pedidos do professor responsável, como o “Contra Corona”.

### Técnica:

A técnica utilizada foi uma adaptação do GDC, as reuniões do SCRUM nos começos de projetos levavam bastante tempo para que os requisitos ficassem muito bem definidos e delimitados, impedindo que os projetos tomassem proporções descabidas para a proposta do “Ludo Educativo” [1].

### Tecnologia Utilizada:

A primeira tecnologia usada foi uma *engine* chamada GameMaker2 [2], nela é possível fazer jogos de forma simplificada e objetiva, ela conta com dois modos de programação, sendo um deles *drag n’drop* e o outro via scripts.

O primeiro projeto chamado “Linha do tempo - Roma” realizado foi utilizado para treinamento, as dúvidas e dificuldades eram tratadas imediatamente (não era necessário esperar a reunião semanal), ao mesmo tempo em que o supervisor ia explicando e expandindo o conhecimento sobre as capacidades da ferramenta.

Após esse primeiro projeto foi adquirida uma noção muito boa com relação aos problemas podem ser evitados com uma boa organização ao se fazer o GDC. A linguagem que o GameMaker2 [2] utiliza é muito próxima de C, facilitando o aprendizado da mesma sem grandes dificuldades com relação à linguagem.

No último projeto chamado “Contra Corona” foi utilizada a ferramenta Unity3D [3], sendo mais conhecida e com grande suporte da comunidade (novamente a linguagem em si não foi o maior desafio, visto que é possível fazer os scripts utilizando C#). A *engine* é extremamente completa e intuitiva (a facilidade de adaptação ocorreu devido ao convívio com o GameMaker2 [2], pois a Unity3D [3] contém várias facilidades e atalhos que poderiam ser úteis para enriquecer e facilitar o uso da GameMaker2 [2]).

Para esta ferramenta também houve um treinamento via projeto, este projeto foi o “Contra Corona”. O método foi ligeiramente diferente do anterior, pois as noções básicas de organização a respeito de jogos já estavam mais desenvolvidas, ainda havia o suporte constante, porém, foi menos necessário.

Nenhuma das *engines* foram ensinadas durante a graduação, porém os conceitos necessários para sua utilização foram, como programação orientada a dados, engenharia de software e basicamente todas as matérias relacionadas à programação do curso de engenharia de computação, inclusive conceitos tidos como básicos foram utilizados extensivamente por serem de alta eficiência, como listas e dicionários (em conjunto com estruturas próprias da plataforma). Dito isso, o treinamento na ferramenta foi suficiente para desempenhar o papel proposto do estágio, que era o desenvolvimento de jogos educacionais.

## 5) Projetos em que o aluno atuou durante o semestre

Cada projeto abaixo conta com uma publicação no Facebook (caso já tenha sido publicado), sua descrição será feita seguida dos desafios encontrados durante a execução do projeto.

### Linha do tempo – Roma (GameMaker2 [2])

##### Objetivos

Criar um jogo para reforçar o que foi estudado pelos alunos do ensino fundamental a respeito dos acontecimentos importante na Roma antiga.

O projeto do jogo foi concebido como um jogo do estilo *Timeline* onde os jogadores recebem os eventos temporais do assunto do jogo de forma aleatória.

O objetivo de cada jogador no jogo é esvaziar a mão, para isso ele não pode errar a posição de seus eventos, pois a cada carta usada ocorre uma verificação, se ela estiver correta sua data é exibida e ela é fixada na linha do tempo, caso contrário sua data também é exibida mas ela é fixada no lixo.

É possível jogar em até 3 jogadores, eles competem entre si para esvaziar a mão primeiro, porém também é possível jogar sozinho, nesse caso o jogador deve tentar conseguir o maior número de pontos, para isso existe o sistema de combo de pontuação.

##### Desafios

Como primeiro projeto o maior desafio foi a adaptação a uma plataforma comercial, o GameMaker2 [2] funciona na base de eventos, cada objeto pode tem até 15 eventos independente dos eventos padrões (interações de colisão e IO por exemplo).

Outro desafio muito grande foram as decisões de design e UX, como foi o primeiro projeto em que foi necessário cuidado com a idade dos usuários (crianças do ensino fundamental), foi tomado cuidado redobrado para que os botões não ficassem pequenos demais, ou os textos explicativos dos eventos ficassem com vocabulário desnecessariamente complicados.

##### Resultados

O resultado pode ser conferido no link abaixo, é possível jogá-lo em Android e IOS além da versão disponível no “Ludo Educativo” [1].

Web:

<https://www.ludoeducativo.com.br/pt/play/linha-do-tempo-roma>

Android:

<https://play.google.com/store/apps/details?id=com.br.ludoeducativo.cards>

IOS:

<https://apps.apple.com/br/app/linha-do-tempo-roma/id1489350403>

### Vacinax, AlfaBeta Herói, Labincrível, Resgatinhos, Quem Come O Quê (GameMaker2 [2])

##### Objetivos

Subir builds desses jogos já existentes e prontos na loja da Apple.

##### Desafios

Apesar de já estarem funcionando e sem bugs, a mudança da plataforma alvo gerou alguns problemas, como formato de tela, modo de abrir links externos e outras coisas pequenas, mas que impediam o jogo de funcionar perfeitamente. Estes problemas foram resolvidos com ocasional ajuda do supervisor.

##### Resultados

O resultado desse trabalho foi a presença na loja da Apple, gerando mais visibilidade para o “Ludo Educativo” [1], quanto mais visibilidade mais acessos e reconhecimento da plataforma pelos alunos e professores.

Os links se encontram abaixo:

Quem come o quê:

<https://apps.apple.com/br/app/quem-come-o-qu%C3%AA/id1455668146>

Resgatinhos:

<https://apps.apple.com/br/app/resgatinhos/id1372216929>

Labincrível:

<https://apps.apple.com/br/app/labincr%C3%ADvel/id1485910990>

AlfaBeta Herói:

<https://apps.apple.com/br/app/alfabeta-her%C3%B3i/id1455528643>

Vacinax:

<https://apps.apple.com/br/app/vacinax/id1455627184>

### Ludo Simulado (Android Studio)

##### Objetivos

Esse projeto foi criado com o objetivo de levar questões de simulado de forma mais acessível aos usuários (o público alvo era o ensino médio).

##### Desafios

Auxiliar o responsável pelo projeto com resolução em tarefas menores para que ele pudesse focar nas funcionalidades.

##### Resultados

O resultado pode ser conferido no link abaixo, o lançamento foi um grande sucesso para a área, abrindo a oportunidade do projeto seguinte (MetA-SP).

<https://play.google.com/store/apps/details?id=br.com.ludoeducativo>

### MetA-SP (Android Studio)

##### Objetivos

Os objetivos desse projeto incluíam a diminuição no uso de papel na aplicação das provas AAP, pois elas são aplicadas várias vezes ao ano em todos os anos e em todas as escolas de ensino fundamental e médio da rede pública do estado de São Paulo.

##### Desafios

O maior desafio foi continuar auxiliando e programando pequenas partes como tutorial e balões de confirmação de forma coesa com o código escrito anteriormente pelo programador responsável pelo projeto.

Outro desafio foi implementar as funcionalidades e *fixes* à tempo da próxima aplicação, sempre um período aproximado de 2 meses.

Ainda não relacionado à programação, mas igualmente desafiador foi ensinar os professores e responsáveis pela aplicação das provas como o sistema funciona.

##### Resultados

O resultado foi bastante animador, visto que o projeto foi aplicado várias vezes e ainda expandido para uma versão Web (para acessar é necessário usuário e senha por conter informações sensíveis sobre provas e gabaritos).

A versão em Android pode ser conferida no link abaixo:

<https://play.google.com/store/apps/details?id=br.com.ludoeducativo.redeeducacao>

### Elementar (GameMaker2 [2])

##### Objetivos

Produzir o primeiro jogo do ludo com interação entre jogadores, cada usuário no site tem um número que podemos usar como *tag* para transmitir mensagens, no caso desse jogo são transmitidas cartas.

O jogo consiste em conseguir o maior número de cartas, cada carta se refere à um elemento da tabela periódica, cada carta contém informações a respeito do elemento.

O jogador pode ter mais de uma mesma carta, sendo possível trocar as repetidas para conseguir as faltantes. Para estimular a troca entre jogadores colocamos peso na distribuição das cartas de forma aleatória, a fim de que cada jogador terá mais dificuldade em conseguir um determinado grupo de elementos (sendo mais fácil conseguir por meio de trocas).

##### Desafios

Interagir com o banco de dados da Aptor para ler as questões (presentes no jogo como forma de bônus) e manter a coerência da troca de mensagens.

Essa interação com o servidor não é feita de forma facilitada no GameMaker2 [2], foi necessário usar plugins e ferramentas externas (como manipulação de HTML, AJAX e outros) para que o funcionamento ocorresse de forma adequada.

##### Resultados

Este jogo foi feito sob o pedido direto do professor responsável, toda a parte da jogabilidade e troca está implementada e funcionando, estamos apenas aguardando a criação da parte artística das cartas com suas informações (o jogo precisa ter informações precisas, o público alvo dessa vez é o colegial e conta com questões bônus de vestibular).

O link para acesso antecipado se encontra abaixo:

<https://www.ludoeducativo.com.br/pt/play/elementar>

### Dominenglish (GameMaker2 [2])

##### Objetivos

Dar continuidade no projeto do estagiário anterior e postar. No momento os bugs já foram corrigidos e só falta mudanças na parte de UX e adição de sprites e outras coisas da parte artística.

##### Desafios

Continuar um projeto que não foi de autoria original minha, foi necessário usar toda a experiência obtida previamente para conseguir entender e dar continuidade ao projeto de forma eficiente.

##### Resultados

Ainda há algumas coisas artísticas a serem adicionadas, porém o jogo é jogável (sua lógica e *game loop* funcionam de maneira correta).

### O dever me chama (GameMaker2 [2])

##### Objetivos

Como o jogo foi totalmente desenvolvido fora da área e somente iriamos postar nas lojas da Google, Apple e no “Ludo Educativo” [1], somente foi necessário fazer pequenas adaptações e sugestões de UX.

##### Desafios

Usar a experiência obtida anteriormente para guiar outros no uso da mesma plataforma.

##### Resultados

O jogo está funcionando corretamente e já está pronto para lançamento, somente estamos esperando a data combinada para realizar essa operação.

Os links para acesso estão abaixo:

Web:

<https://www.ludoeducativo.com.br/pt/play/o-dever-me-chama>

Android:

<https://play.google.com/store/apps/details?id=br.com.ludoeducativo.odevermechama>

IOS:

<https://apps.apple.com/br/app/o-dever-me-chama/id1491809591>

### Contra Corona (Unity3D [3])

##### Objetivos

Reforçar a importância das medidas profiláticas em meio a pandemia do COVID-19, o jogo é composto por três mini games, cada um com um objetivo e explicação ao iniciar (juntamente com um tutorial que pode ser pulado).

##### Desafios

Utilizar a Unity3D [3] em tempo hábil, primeiro projeto com 5 pessoas envolvidas (uma artista, um músico, três desenvolvedores). Além de ter sido o primeiro projeto em Unity3D [3] da área também foi o primeiro projeto a ter tantas pessoas envolvidas.

##### Resultados

O jogo foi completo e está em vias de ser postado na loja da Google, ele já pode ser acessado no “Ludo Educativo” [1] por meio do link abaixo, também é possível conferir sua menção numa página da FAPESP (link abaixo).

Web:

<https://www.ludoeducativo.com.br/pt/play/contra-corona>

FAPESP:

<http://agencia.fapesp.br/ludo-educativo-lanca-game-contra-corona/33266/>

## 6) Descrição das Atividades Realizadas

Nessa sessão as atividades serão separadas por semestres, sendo eles três desde a contratação.

### Primeiro semestre

Nesse período foram contemplados dois projetos principais, o “Linha do tempo - Roma” e o “Ludo Simulado” [3], o segundo sendo de menor participação e o primeiro sendo de minha responsabilidade.

Para o primeiro projeto houve um planejamento que consistiu em definir o estilo de jogo (mecânicas baseadas em jogos reais), obter jogos e mídia de referência (podendo ser jogos eletrônicos ou vídeos e imagens relacionadas), definição de requisitos do jogo (limitando os modos de jogo e definindo como o jogador seria recompensado ou penalizado), programação e aprendizado do GameMaker2 [2] e por último a publicação.

Para o segundo não houve participação direta na programação do projeto, mas sim nas decisões do projeto durante as reuniões do SCRUM adaptado.

No “Linha do tempo - Roma” o estilo de jogo foi concebido com a ajuda de todos da equipe por se tratar do meu primeiro projeto, seria necessário que fosse algo simples de implementar, mas que permitisse o aprendizado e utilização de boas práticas de programação. Foi escolhido então um jogo de cartas por haver muitas referências de jogos de cartas e todos conhecerem como cartas devem se comportar (ordem em que devem aparecer, movimentação e manipulação).

Definido que seria um jogo com a utilização de cartas foi discutido primeiro qual tipo de jogo seria interessante para se acrescentar conceitos aprendidos em sala, o tipo de jogo *Timeline* se mostrou muito bom para jogos em que existe uma ordem ou sequência que devem ser respeitados. Escolhido o estilo *Timeline* restava escolher que matéria e qual parte dessa matéria seria contemplada no jogo, o conteúdo escolhido foi o período da história da Roma antiga, muito abordado nos materiais escolares e presente nas diretrizes de educação pública como matéria obrigatória.

Após escolher sobre o que seria o jogo era necessário aprender como um jogo desse tipo é feito, jogos de referência e vídeos de pessoas manipulando cartas foram muito importantes para que as cartas do jogo tivessem a movimentação correta (ou mais próxima do natural), as referências eram observadas e obtidas ao mesmo tempo em que o código do projeto começava a ser feito, um repositório no Gitlab foi criado para este fim.

O GameMaker2 [2] separa as partes de um jogo por *room*, por exemplo, um menu pode ser uma *room* e uma parte com *gameplay* pode ser outra, ou ainda, cada fase de um jogo pode ser uma *room*. Com isso em mente o projeto se iniciou na *room* do jogo, onde foram testadas mecânicas de *drag n’ drop*. Para fazer os testes usei *sprites* simplificados que continham apenas o formato de uma carta, mas nada verossímeis ao produto final, esses *sprites* são conhecidos como *mockup*, servem para o desenvolvedor ter algo com o que trabalhar e testar seu código enquanto o artista responsável desenvolve as imagens e animações finais.

Durante o desenvolvimento foram utilizadas as estruturas de **dicionário** e **listas** de forma extensiva. O jogo tem regiões que tem cartas, elas são as mãos dos jogadores, o próprio *Timeline*, o lixo e as cartas que ainda não foram dadas. Cada região tem uma lista que diz quais cartas estão nela, e operações de **adição** e **remoção** são realizadas constantemente. A leitura das cartas (data em que deve ser inserida, texto informativo curto, texto informativo longo e ID da imagem relacionada) é feita a partir de uma planilha (depois da experiência desse projeto percebi a necessidade do uso do JSON) que é armazenada numa estrutura semelhante à um dicionário.

O GameMaker2 [2] não conta com ferramentas de animação própria (como manipulação de transformação de objeto), porém a Aptor tem uma biblioteca feita chamada *Anima*, ela contém funções primitivas para controle de movimentação, com a possibilidade de uso de *easing*, essa biblioteca permitiu a movimentação de forma bem completa, resultando num jogo mais fluido.

Para que a *Anima* funcione corretamente é necessário o uso de máquinas de estado para que os objetos animem de acordo com o estado em que se encontram. O conceito de máquinas de estado também foi utilizado para a controladora do jogo.

Nas animações a máquina de estado é utilizado para controlar quais ações serão tomadas e qual animação deve rodar, encadeando animações ou então sendo o gatilho para operações no jogo. Já a controladora do jogo foi uma ótima prática de programação que torna todo o desenvolvimento e lógica do jogo de forma coesa, diminuindo a chance de erros e inconsistências no código e *game flow*.

Tendo a lógica implementada e funcionando bem (com os menus e controles de som e opções do jogo) foi possível adicionar a arte do jogo, com os ajustes necessários nas animações e interações do jogador o jogo já estava praticamente completo.

Antes de publicar o jogo foram feitos testes exaustivos para encontrar bugs e inconsistências, para testar as mecânicas do jogo e o comportamento dele com os mais variados tipos de interação (desde não fazer nada até fazer tudo de forma fora da esperada), esses testes ajudaram a encontrar problemas no jogo que geralmente se encontravam na máquina de estados que não estava preparada para receber entradas ditas como inválidas.

Depois dos vários testes foi possível publicar, na plataforma do “Ludo Educativo” [1] a publicação se dá de forma simplificada por meio de *iframe*, já nas lojas da Google e Apple existem vários requisitos que devem ser preenchidos, desde checagem de conteúdo até faixa etária alvo. Essas particularidades foram exigidas de forma diferente entre as lojas, mas fazem parte dos contratos entre desenvolvedor e plataforma.

Com a publicação nas plataformas de consumo por último e muito importante foi feito um post no facebook do “Ludo Educativo” [1], para que os seguidores conheçam a novidade da equipe.

A participação no “Ludo Simulado” se concentrou na parte final do projeto, onde foram realizados testes para verificar a consistência do aplicativo, e publicação na loja da Google. Como esse aplicativo foi feito em Android nativo uma versão para IOS precisaria ser refeita do zero.

A parte de testes foi muito interessante, pois o aplicativo entra em contato com a base de dados da Aptor para baixar as perguntas e imagens, para que o processo seja feita de forma eficiente e não seja limitado pela banda do servidor foi necessário encontrar uma forma de transferir de forma compacta as questões com as imagens (quando presente).

### Segundo semestre

Os principais projetos desse semestre foram o “MetA-SP” e o “Elementar”, sendo a maior participação no “Elementar”, a responsabilidade do projeto foi atribuída à dois estagiários devido à escala do projeto ser maior que a média, a participação do segundo estagiário teve início no meio do projeto.

Este projeto foi concebido de forma diferente dos outros, houve um pedido de um professor da UFSCAR para que este contivesse interação entre os jogadores de alguma forma, e que a recompensa do jogador fosse relacionada aos elementos químicos e a completude da tabela periódica.

Para tanto a área se reuniu várias vezes com ideias de projetos que foram apresentados aos “clientes” do jogo, apresentamos as limitações da plataforma o que poderia ser feito para obter um jogo que tivesse os requisitos de forma adequada. Após as reuniões ficou acordado que o jogador deveria passar por 4 estágios se quisesse completar da forma mais eficiente possível sua tabela periódica.

Os quatro estágio são: uma sequência de quatro perguntas que podem ser usadas de bônus na fase de *puyo*, mini game *puyo puyo* onde o jogador pode usar os bônus conseguidos na fase anterior para consumir um elemento (carta) e de acordo com a raridade a ele atribuída um multiplicador será adicionado, uma tabela periódica com informações sobre os elementos descobertos, e no final a loja, onde o jogador pode comprar pacotes de cartas de acordo com suas raridades (a chance de vir um elemento mais raro muda, mas ainda é aleatório) e pode trocar com jogadores.

Estágio de perguntas:

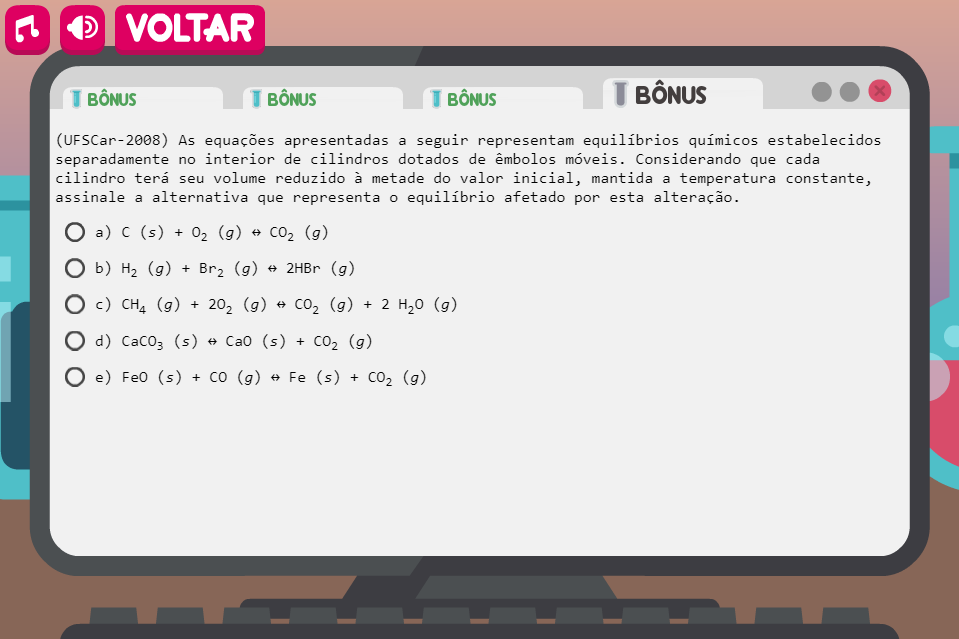


Figura 1 – Tela de perguntas e respostas – Elementar.

Estágio do *puyo puyo* (seleção de bônus e jogo):



Figura 2 – Tela de bônus – Elementar.

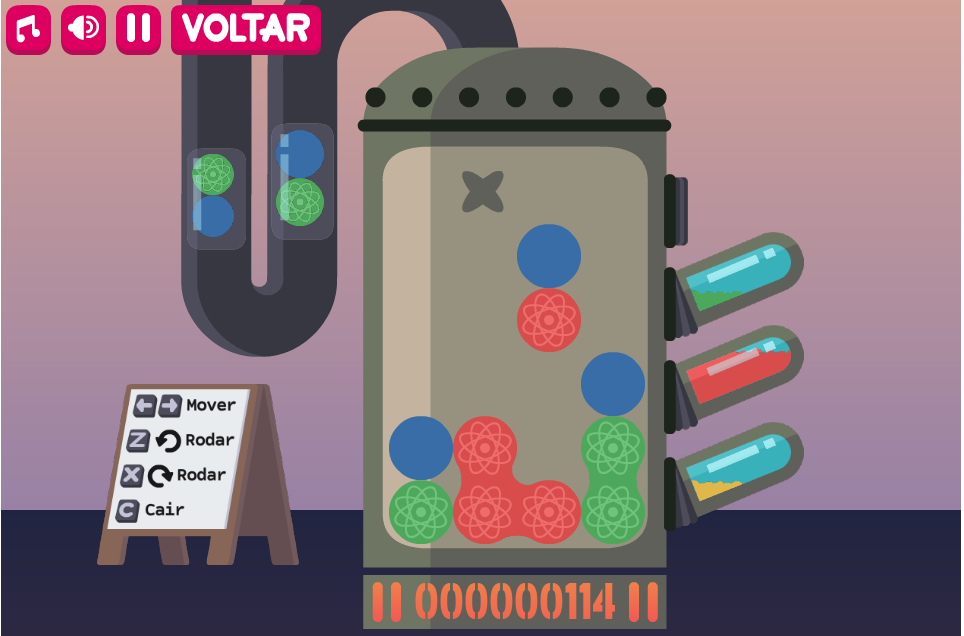


Figura 3 – Tela de mini game *puyo puyo* – Elementar.

Estágio da tabela (com apenas alguns elementos encontrados e informações detalhadas):



Figura 4 – Tela da tabela periódica – Elementar.



Figura 5 – Tela de informações – Elementar.

Estágio da loja (compra e troca de elementos):



Figura 6 – Tela da loja – Elementar.



Figura 7 – Tela de compra – Elementar.



Figura 8 – Tela de recebidos – Elementar.



Figura 9 – Tela de envio – Elementar.

Tirando a arte das cartas dos elementos e as informações neles contidas, toda a arte foi feita pela área de games, havia a opões das informações e imagens dos elementos serem feitos dentro da equipe, porém a pedido do professor essa parte foi feita por eles.

Para esse projeto foram utilizados os mesmos conceitos utilizados no primeiro projeto, máquinas de estado, controladoras, estrutura de dados, e outras particularidades do próprio GameMaker2 [2].

Além das capacidades expandidas a partir do primeiro projeto esse também contou com a necessidade de versionamento e uso de *branches*, prática comum para quaisquer projetos conjuntos, portanto a organização foi de extrema importância.

No momento deste relatório ainda estamos aguardando a arte e informações dos elementos restantes (temos até o *Americium* - 95).

A publicação desse jogo somente poderá ser feita na plataforma do “Ludo Educativo” [1], por ter textos científicos com subscrito e sobrescrito nas fórmulas foi necessário usar um plugin para que sua exibição fosse feita por HTML.

Para o projeto do “MetA-SP” foram realizadas várias iterações, sendo a primeira a adaptação do “Ludo Simulado” para atender aos requisitos exigidos pelo professor responsável. Estes eram a capacidade de fazer a prova durante tempo indeterminado (antes havia o limitador de tempo para que os alunos treinassem para a situação de vestibular), não mostrar a resposta correta (antes o aplicativo mostrava a correta caso a resposta escolhida estivesse errada), navegação diferente entre as questões e uma plataforma separada para que os professores colhessem os resultados das provas.

Durante o “MetA-SP” a área de games teve contato com a D.E. de São Carlos para auxiliar na aplicação das provas, desde ir presencialmente auxiliar na organização e uso da aplicação até a reunir com os responsáveis e seus representantes pela aplicação na cidade.

### Terceiro semestre (atual)

Os principais projetos desse período foram “O dever me chama” e “Contra Corona”, sendo o primeiro o primeiro projeto que eu auxiliei outros no uso da ferramenta e o segundo o primeiro contato com Unity3D [3] na empresa.

“O dever me chama” é um jogo plataforma em que o jogador precisa utilizar uma mecânica de pulo para conseguir desviar dos obstáculos e chegar aos objetivos (coletar os lixos e recicla-los ao final da fase). O conceito do jogo e suas mecânicas foram feitas fora da área de jogos e foi meu trabalho inspecionar e ajudar a corrigir bugs e ajustes de UI para que ele ficasse jogável em plataformas mobile da mesma forma que Web.

Já o segundo projeto, “Contra Corona”, foi feito a pedido do professor responsável. Porém diferente do “Elementar” aqui houve total autonomia sobre mecânicas e estilos de jogo, o único requisito era que o jogo ajudasse na conscientização de quem jogasse.

Para tanto resolvemos criar um jogo com separação de mini games para que cada medida profilática fosse abordada separadamente, sendo o foco de cada jogo, um programador especialista em Unity3D [3] ajudou os outros dois estagiários a usar a ferramenta em tempo hábil de publicação, sendo um desses dois eu mesmo.

Decidimos abordar três medidas, lavagem de mãos (o jogador precisa realizar os movimentos corretos e durante uma duração mínima de 20 segundos para que consiga a pontuação máxima), não aglomeração de pessoas (nesse as “pessoas” são representadas por *emojis* e elas ficam indo em direção ao ponto médio de suas posições) e o meu mini game onde as pessoas andam aleatoriamente num mercado e você precisa administrar máscaras e impedir espirros para evitar contaminação, “Etiqueta do Espirro” (sempre de forma aleatória uma das pessoas que entram no mercado estará contaminada e não há indicação visual de quem é, assim como na vida real não podemos saber sem fazer o teste).



Figura 10 – Tela de menu inicial – Contra Corona.

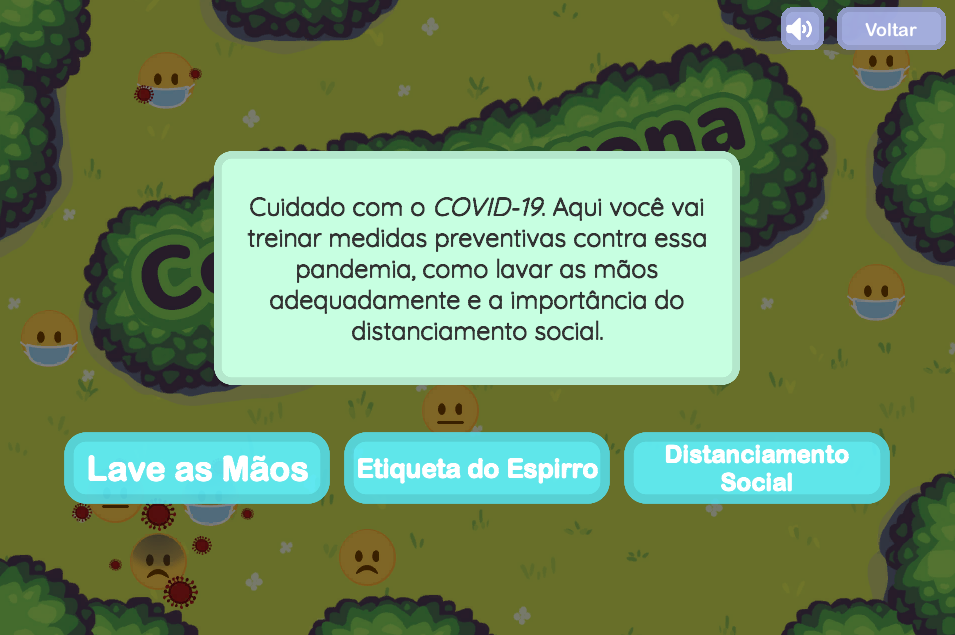


Figura 11 – Tela de escolha do mini game – Contra Corona.



Figura 12 – Tela de jogo – Contra Corona.

Primeiro foi necessário definir o objeto pessoa, a Unity3D [3] tem uma ferramenta própria de animação e estados (*animator* – é uma ferramenta para criar graficamente máquinas de estado com animações e gatilhos que chamam trechos do código quando são atingidos), separando as transições com os estados lógicos para que o jogo consiga contar pontos através do controlador de pontos.

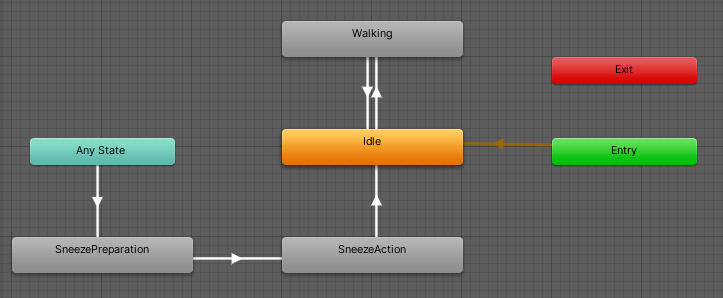


Figura 13 – Máquinas de estado da pessoa – Contra Corona.

Depois foi necessário criar o método de “caminhada” das pessoas, isso foi feito usando o conceito de *waypoint*, onde o objeto em questão se move em direção ao *waypoint* e não é esperado nenhum obstáculo nesse trajeto, a escolha de waypoint depois que as pessoas já estavam na cena era de forma aleatória, sendo o único momento em que havia controle sobre qual *waypoint* seria escolhido era quando a pessoa era instanciada na cena. Elas são instanciadas na entrada do mercado e vão até um *waypoint* específico no meio da cena e de lá a escolha é feita de forma aleatória.

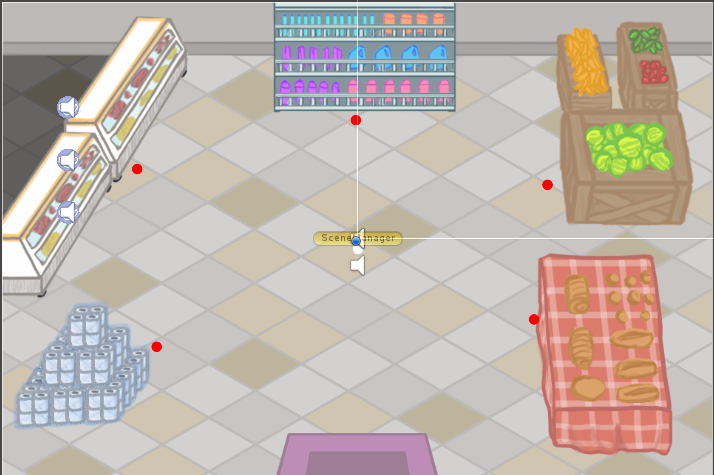


Figura 14 – *Waypoints* do jogo – Contra Corona.

Feita a movimentação e os ciclos de espirro e interrupção (incluso interação com entrada do usuário), fiz a adição do *drag n’drop* das máscaras, foi interessante por perceber que os objetos devem ter uma caixa de colisão maior que a própria imagem para este fim, uma vez que a alta precisão dificulta a interação reduzindo muito a acessibilidade.

A Unity3D [3] utiliza C# para programar os scripts utilizados em jogo, então foi possível e necessário utilizar os conceitos de classe aprendido no curso para que a lógica funcionasse de forma correta, é praticamente impossível fazer um jogo sem a orientação a objetos (a coesão proporcionada pela POO é extremamente valiosa para que o programa funcione, classes, interfaces, tipos de classe, hierarquia e sobrecarga foram presentes durante todo o projeto e em todos os mini games).

Em contraste ao GameMaker2 [2] a Unity3D [3] expõe o uso de classes e permite uma customização muito maior, melhorando o desempenho e organização do projeto (e a legibilidade como consequência).

## 7) Dificuldades e Limitações

A maior dificuldade foi conciliar o estágio com a vida acadêmica, uma vez que eu ainda tinha matérias a serem realizadas e o estágio acabava por vezes roubando um pouco meu foco (fazer jogos é uma paixão pessoal e tenho uma profunda ligação familiar com eles, fizeram parte da minha infância e muitas memórias importantes estão ligadas a isso). Esse esforço em manter o foco de forma igual e priorizar as aulas e provas ajudou muito no crescimento pessoal, acredito que essa habilidade de conciliar as obrigações será útil não somente na minha vida profissional, mas principalmente na pessoal.

Dificuldades como introdução a novas tecnologias e formas de programar acabaram por não ser tão limitadoras, pois era muito interessante e divertido conseguir fazer coisas cada vez mais complexas nas plataformas utilizadas.

O maior limitador foi o tempo de produção, como a proposta do “Ludo Educativo” [1] são jogos curtos e simplificados é necessário que a plataforma sempre tenha alguma novidade, a fim de manter os usuários curiosos e interessados, gerando acessos e fornecendo visibilidade ao Ludo e à Aptor. Dito isso considero que a limitação de tempo na verdade ajudou a melhorar o método de produção de jogos (programação à parte), como discussão do projeto, melhorias de design e quaisquer outras coisas que podiam ser melhoradas para aumentar a eficiência na comunicação entre os envolvidos da área.

Acredito que a maior lição aprendida seria a da necessidade de comunicação clara com o time, sem isso não seria possível aprender a usar as ferramentas da forma correta, ou tomar as decisões de projeto na direção correta. Com uma boa comunicação as limitações pontuais de cada um acabam se tornando menores, já que os outros estão cientes delas e podem ajudar a superá-las ou auxiliar com formas alternativas de resolver os mais diversos problemas.

## 8) Contribuições do estágio para o aluno

Participar do estágio reforçou a importância da pontualidade de entrega, dedicação ao que se é proposto e responsabilidade com todos os envolvidos. Independente do objetivo que um projeto possa ter, é muito improvável que alguém o realize sozinho, sempre haverá uma equipe por trás, por mais que possa existir um responsável pelo projeto em questão, portanto é sempre necessário ajudar e fazer o que é possível para que os objetivos comuns sejam atingidos.

Além da experiência profissional, presenciei o crescimento pessoal dos colegas e das dificuldades que cada um enfrenta no dia a dia, essa exposição e preocupação com cada um me ajudou a exercitar uma empatia que eu não achei que seria possível ter num ambiente de trabalho.

## 9) Relacionamento entre o Curso e o Projeto de Estágio

As disciplinas diretamente relacionadas a programação (desde introdução à ciência até engenharia de software) contribuíram para o desenvolvimento do estágio, acredito que a produção de jogos utiliza vários aspectos vistos durante o curso. Porém os cursos que não contribuíram muito foram os relacionados à elétrica e eletrônica (sou estudante da engenharia de computação e acabei não usando muito essa parte).

## 10) Considerações sobre o Curso de Graduação

Para a engenharia de computação eu acredito que a proporção de matérias relacionadas à computação podia ser maior (considerando as obrigatórias), com disse anteriormente, acredito que produzir jogos utilize muitos conhecimentos de computação e que isso poderia ser mais abordado na graduação, não de forma velada, mas de forma clara como as necessidades de se fazer um programa que seja eficiente e que tenha uma boa lógica por trás.

Frequentemente meus colegas de trabalho comentavam sobre assuntos relacionados à Web, para a engenharia essas matérias não são obrigatórias, porém se o estágio fosse em algo voltado à essa parte eu teria usaria muito mais tempo na adaptação (mesmo tentado usar recursos como Flask e interfaces em navegadores quando possível).

## 11) Trabalhos Futuros

No momento deste relatório estou responsável por um projeto que envolve o ensino de geologia para alunos do fundamental, a ferramenta escolhida foi a Unity3D [3] e gostaria de termina-lo para ter a máxima contribuição com a plataforma “Ludo Educativo” [1], acredito que há muito potencial a ser explorado nessa área da empresa, basta apenas que os recursos e esforços corretos sejam aplicados.

Gostaria de continuar trabalhando com a produção de jogos, sei que é um privilégio trabalhar com o que gosto tanto, sei também que a área da computação é uma das poucas que realmente pode continuar funcionando remotamente e que isso é outro privilégio da área. Por isso gostaria de contribuir com a sociedade da forma que considero mais interessante, com jogos que possam passar alguma mensagem importante, seja ela educacional ou social.

## 12) Referências (opcional)

|  |  |
| --- | --- |
| [1] | Aptor Software, “Ludo Educativo,” [Online]. Available: https://www.ludoeducativo.com.br/pt/. |
| [2] | YoYo Games, [Online]. Available: https://www.yoyogames.com/gamemaker. |
| [3] | INSTITUTO CLARO, “Vestibular: game gratuito prepara estudante para exames e pode ser usado por professor em sala de aula,” 14 Maio 2019. [Online]. Available: https://www.institutoclaro.org.br/educacao/nossas-novidades/podcasts/vestibular-game-gratuito-prepara-estudante-para-exames-e-pode-ser-usado-por-professor-em-sala-de-aula/. [Acesso em 31 Maio 2020]. |