Relatório de Projeto Análise de Sistemas

(2020/2021)

Grupo Nº	Curso / Nome Projeto
3	Engenharia Informática - TCP Tactical Combat Planner

Composição do Grupo					
Número / Nome	Esforço (Horas)				
	Pesqui. Web	Reuniões	Elabor. Diag.	Elabor. Relató.	Total
50039473 Jaime Ferreira	20	12	30	10	72
50039077 Fernando Lamar	20	12	32	8	72

Versões do Relatório

Versão	Data	Autor	Descrição
1.0	01/11/20	Jaime Ferreira	Versão inicial dos requisitos Funcionais
1.1	15/11/20	Fernando Lamar	Especificação de API Versão final dos requisitos Funcionais, Requisitos não funcionais Diagrama de casos de utilização
1.2	13/12/20	Fernando Lamar	Modelo de Domínio Diagrama de contexto Descrição dos casos de uso
1.3	27/01/21	Jaime Ferreira	Versão Final
1.4	08/02/21	Jaime Ferreira / Fernando Lamar	-Alteração no diagrama de contexto e na descrição do mesmo -Alteração no diagrama use cases .foi implementado 2 uses cases novos des da ultima entrega -Alteração no diagrama modelo de dominio .Fix em metodos e atributos na classe EventGroup -Alteração no anexo D: .Descrição das APIs que facilitam a interação dos stakeholders com o nosso SOI -Alteração no diagrama de componentes(BPMN) .tinhamos mais do que um input no subprocesso, mudamos o subprocesso para ter loop e pusemos em comentario a condição de saida.

índice

Sumário Executivo 3
Acrónimos 3
1 Introdução 4
2 Diagrama de Contexto do Sistema TCP 5
3 Diagrama de Casos de Utilização do Sistema TCP 6
3.1 Casos de Uso – Descrição Geral 6
UC01 – Criar Equipa 6
UC02 – Convidar Pessoa para uma equipa 7
UC03 – Criar Evento 7
UC04 – Procurar Eventos 7
UC05 – Procurar Campos 7
UC06 – Atribuir Cargos 7
UC07 – Receber Notificações do Evento 7
UC08 – Criar Tácticas 7
UC09 – Confirmar Presença num Evento 7
UC10 – Visualizar Representação 2D 7
UC11 – Visualizar Representação 3D (Não implementado) 7
UC12 – Visualizar Evento 7
UC13 – Solicitar Inscrição em Evento 7
UC14 – Convidar pessoas para Evento 7
UC15 – Alterar definições do Evento 7
UC16 – Convidar jogador existente para Equipa 8
UC17 – Visualizar Táticas 8
UC18 – Alterar definições de Equipa 8
UC19 – Definir/alterar grupos 8
UC20 – Desenhar Elementos 8
UC21 – Criar Subgrupos (Não implementado) 8
UC22 – Atribuir Ranking 8
3.2 Descrição Detalhada dos Casos de Utilização 8
UC01 - Criar Equipa 8
UC03 - Criar Evento 9
UC08 - Criar Táctica 9
4 Modelo de Domínio do Sistema TCP 10
5 Diagrama Máquina de Estados TCP 12
6 Apresentação da Arquitetura do Sistema 13
7 Especificação de Processos Negócio 15
Criação de Tática: 15
Processo 1: «Criação de Tática» 16
Criação de Evento: 16
Processo 2: «Criação de Evento» 17
8 Conclusões e Trabalho Futuro 18
Referências 19
Biografia dos Autores 19
Anexo A: Manual de Utilização da Aplicação TCP 20
Anexo B: Frameworks utilizadas no desenvolvimento da solução 22
Anexo C: Levantamento de Requisitos do Sistema TCP 23
Requisitos Funcionais 23
Requisitos Não Funcionais 24
Anexo D: Especificação das Interfaces API 24
Mapbox (GPS) 24
Heroku 25
Express+Node.js 25
Anexo E: Funcionamento da Unidade Curricular 26

Sumário Executivo

O Tactical Combat Planner é uma aplicação web, enquadrada no âmbito de entretenimento/jogos de combate, mais especificamente airsoft. Tem como objetivos principais fornecer aos jogadores de airsoft uma plataforma onde podem: criar equipas, criar eventos e fazer táticas para os seus campos de jogo.

Esta aplicação tem o potencial para vir a servir como uma plataforma de referência para o encontro de jogadores na mesma área para formar equipas amadoras e organizar eventos de airsoft. No entanto, o elemento diferenciador desta aplicação para outras é a funcionalidade de criação de táticas para os campo de jogo, onde os utilizadores podem definir movimentações, bases, funções específicas para cada membro, entre outros.

Este relatório pretende criar uma imagem geral do projeto de forma a agilizar o ganho de informação sobre o projeto ao leitor.

Palavras Chave: jogos de combate; airsoft; eventos; equipas; táticas

Acrónimos

SoI	*System of Interest*
TCP	*Tactical Combat Planner*
GUI	*Graphical User Interface*
API	*Application Programming Interface*

1 Introdução

O objetivo da nossa aplicação é oferecer uma plataforma onde os utilizadores podem criar táticas. Escolhemos como público alvo o Airsoft, porque simulam ações e combates militares de uma forma desportiva, nesta área o reconhecimento do campo de batalha é sempre uma vantagem a ter para um maior sucesso.

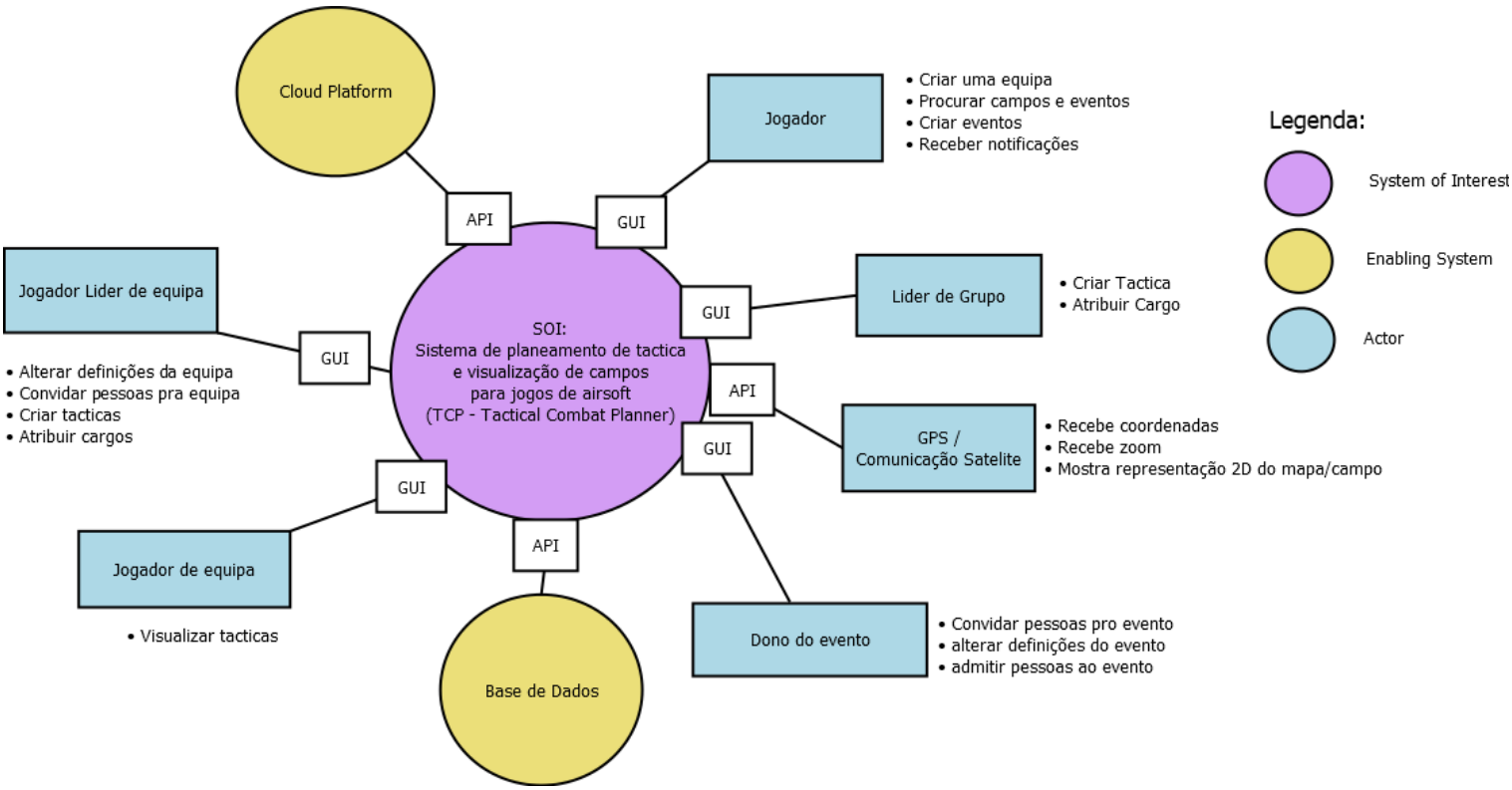
Para criarmos táticas precisamos de um lugar para as guardar, onde apenas pessoas que nós damos permissões podem aceder por isso decidimos criar a funcionalidade Equipas, onde poderemos convidar/remover membros da equipa e atribuir cargos relacionados com airsoft (por exemplo: Sniper, Scout, Riffleman, etc.) para uma melhor personalização destes membros para conseguir mais facilmente atribuir ordens a estes cargos dentro das táticas.

Semelhantemente às equipas, também decidimos criar as funcionalidades Evento, um lugar onde utilizadores individuais ou equipas possam juntar-se e preparar para um jogo entre si. No evento temos definições extras como data, duração, número de grupos, número de jogadores por grupo, com estas duas últimas definições podemos dividir os utilizadores inscritos no evento entre grupos, poderão ser por exemplo duas equipas cada uma com 20 jogadores ou 10 equipas cada uma com 100, tudo isto é definido pelo criador do evento.

Para estas duas funcionalidades funcionarem mais facilmente, decidimos criar a funcionalidade de Convite, assim, um líder de equipa/dono de evento poderá convidar pessoas para as suas equipas/eventos.(nota: um jogador também pode requisitar a entrada numa equipa/evento).

Como o objetivo do projeto é o utilizador poder criar táticas, com estes 3 grupos de funcionalidades ficará muito mais simples a navegação e organização para cumprir estes requisitos. As táticas podem ser criadas para equipas ou para grupos dentro de um evento.

2 Diagrama de Contexto do Sistema TCP



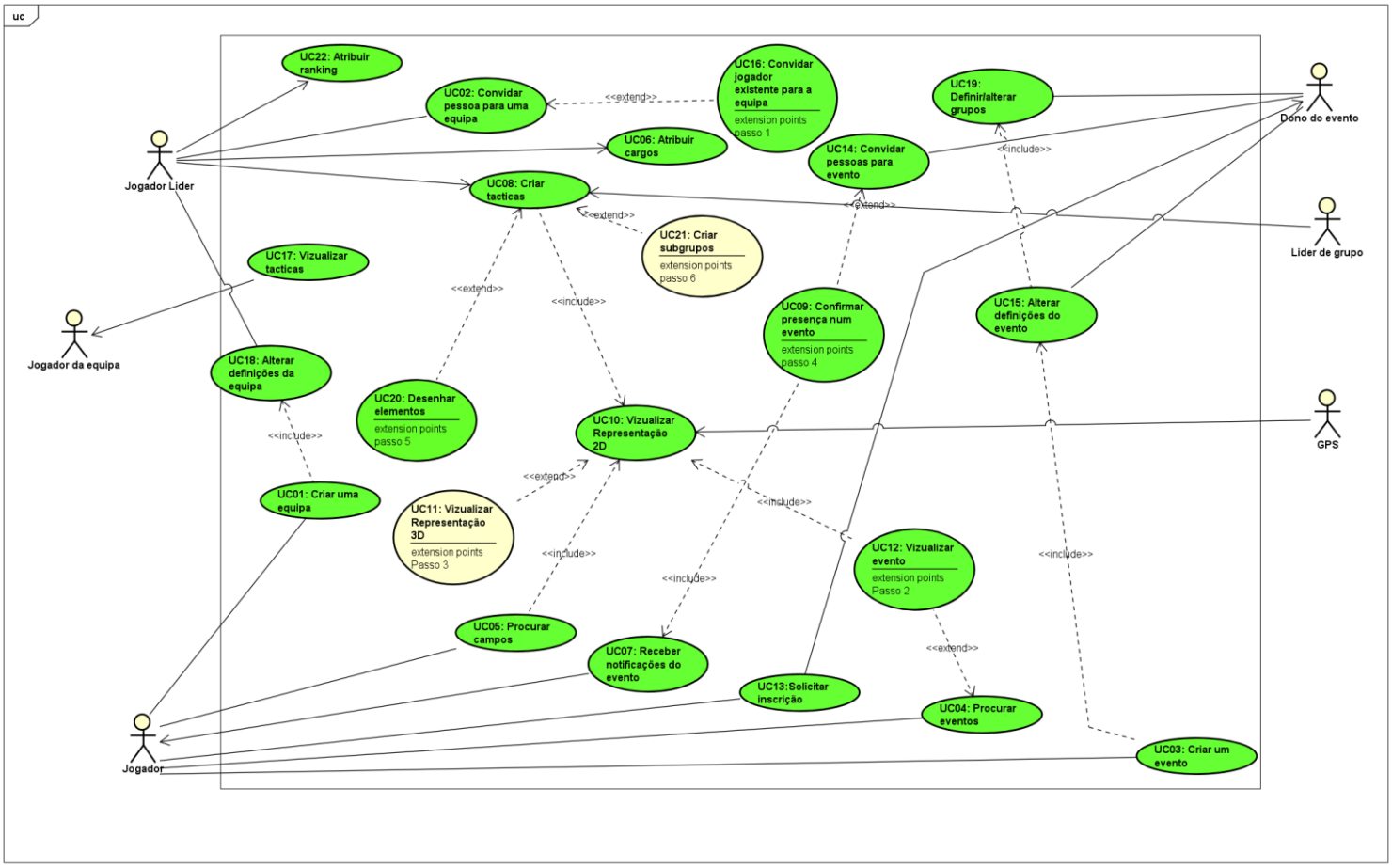
Stakeholders	Role
Jogador	Qualquer utilizador autenticado na app é um ator jogador, este pode criar equipa e eventos, procurar por campos e eventos, solicitar inscrições a eventos e equipas e também receber notificações do sistema.
Jogador de equipa	Este ator tem todas as funcionalidades que o ator Jogador tem. Este ator é membro de uma equipa e poderá ver as táticas previamente criadas.
Jogador lider de equipa	Este ator tem todas as funcionalidades que o Jogador de equipa tem. Este ator pode também criar táticas para as suas equipas, convidar pessoas e alterar as definições da sua equipa.
Dono do evento	Este ator tem todas as funcionalidades que um ator jogador tem. Este utilizador é dono do evento após a criação de um evento, poderá agora convidar pessoas para o seu evento e alterar as definições do mesmo.
Jogador Lider de grupo	Este ator tem todas as funcionalidades que um Ator Jogador tem. Este ator pode atribuir roles e criar táticas para o seu grupo dentro do evento.

GPS	Este ator irá comunicar com a API leaflet. Ao receber coordenadas e zoom, irá mostrar representações do mapa/campo.
Cloud Platform	Será necessário uma plataforma que suporte o nosso sistema, deixando o nosso sistema acessível remotamente online.
Base de Dados	Será necessário um banco de dados onde podemos guardar todos os componentes do modelo de domínio, como por exemplo informação sobre eventos, equipas, táticas, utilizadores etc.

3 Diagrama de Casos de Utilização do Sistema TCP

O diagrama de casos de utilização é o desenvolvimento do anexo C, requisitos funcionais, com os atores que executam os casos de utilização e a descrição do funcionamento de cada um dos casos. Os casos de utilização que estão implementados têm fundo verde e os que são para uma possível implementação futura têm fundo amarelo.

3.1 Casos de Uso – Descrição Geral



UC01 – Criar Equipa

Descrição	*O jogador cria uma equipa, transformando-se em Jogador Líder, podendo assim convidar outros Jogadores, atribuir-lhes cargos e criar tácticas*

UC02 – Convidar Pessoa para uma equipa

Descrição	*O Jogador Líder de equipa pode convidar um jogador para a sua equipa*

UC03 – Criar Evento

Descrição	*O jogador cria um evento, para o criar, define as características do evento*

UC04 – Procurar Eventos

Descrição	*O jogador pode procurar eventos através da lista de eventos*

UC05 – Procurar Campos

Descrição	*O jogador pode procurar campos a partir do mapa*

UC06 – Atribuir Cargos

Descrição	*O Jogador Líder da equipa pode atribuir cargos aos outros membros da equipa (sniper/rifleman etc)*

UC07 – Receber Notificações do Evento

Descrição	*O Jogador após o UC09 começa a receber notificações relacionadas com Evento*

UC08 – Criar Tácticas

Descrição	*O Jogador Líder de equipa cria uma táctica para o mapa selecionado, de forma aos restantes jogadores de equipa poderem visualizar esta táctica e fazerem jogadas coordenadas.*

UC09 – Confirmar Presença num Evento

Descrição	*Após o UC14 ter sido executado pelo Dono do Evento, um convite para o evento é recebido pelo Jogador, e este pode aceitá-lo, tornando-se membro do Evento.*

UC10 – Visualizar Representação 2D

Descrição	*O GPS permite aos Jogadores visualizarem o seu campo em 2D após executarem um dos seguintes casos de uso: UC12, UC05, UC08. Depois disto o utilizador pode executar o UC11.*

UC11 – Visualizar Representação 3D (Não implementado)

Descrição	*Após executar o UC10, o utilizador pode visualizar o seu campo em 3D.*

UC12 – Visualizar Evento

Descrição	*O Jogador pode visualizar o evento que selecionou enquanto realizava o UC04 e por consequência irá executar o UC10*

UC13 – Solicitar Inscrição em Evento

Descrição	*O Jogador pode solicitar inscrição no evento ao Dono do Evento*

UC14 – Convidar pessoas para Evento

Descrição	*O Dono do Evento pode convidar pessoas para o seu Evento.*

UC15 – Alterar definições do Evento

Descrição	*O Jogador ao executar o UC03 executa obrigatoriamente o UC15 para definir as definições de Equipa*

UC16 – Convidar jogador existente para Equipa

Descrição	*O Jogador ao executar o UC02, se o utilizador já estiver inscrito na aplicação, o UC16 é executado.*

UC17 – Visualizar Táticas

Descrição	*Qualquer Jogador da Equipa pode visualizar as Táticas criadas pelo Jogador Líder*

UC18 – Alterar definições de Equipa

Descrição	*O Jogador ao executar o UC01 executa obrigatoriamente o UC18 para definir as definições de Equipa*

UC19 – Definir/alterar grupos

Descrição	*O Dono do Evento depois de executar o UC15, define os grupos do evento, e mais tarde pode alterá-los*

UC20 – Desenhar Elementos

Descrição	*O Jogador Líder ou o Líder de Grupo, depois de de executar o UC08 pode de desenhar elementos na tática*

UC21 – Criar Subgrupos (Não implementado)

Descrição	*O Jogador Líder ou o Líder de Grupo, depois de de executar o UC08 pode criar subgrupos dentro da sua equipa ou grupo para dar movimentações ou indicações a esses subgrupos*

UC22 – Atribuir Ranking

Descrição	*O Jogador pode atribuir Ranking aos membros da sua equipa de forma a poder gerir permissões exemplo: permitir que outro membro possa criar táticas.*

3.2 Descrição Detalhada dos Casos de Utilização

UC01 - Criar Equipa

Descrição	Um jogador cria uma equipa, transformando-se num Jogador Líder, podendo assim convidar outros Jogadores, atribuir-lhes cargos e criar tácticas
Pré-condições	Jogador autenticado com sucesso, ID/nome da equipa não utilizados O Jogador tem de ter clicado na opção equipas no menu principal
Cenário Principal	1. O Jogador clica em criar equipa 2. O sistema mostra uma página com o formulário de criação de equipa 3. O Jogador executa o UC18(Alterar definições da equipa). 40. O sistema faz uma query à base de dados para confirmar o nome de equipa está disponível 5. O sistema cria a equipa com sucesso
Cenário Alternativo	N/A
Pós-Condições	A equipa é adicionada à base de dados O Jogador transforma-se em Jogador Líder

Cenário de Exceção	4.1 O sistema recebe informação que o nome desta equipa já está em uso
Pós-Condições	4.1.1 O sistema avisa o jogador que não poderá proceder com estas definições.

UC03 - Criar Evento

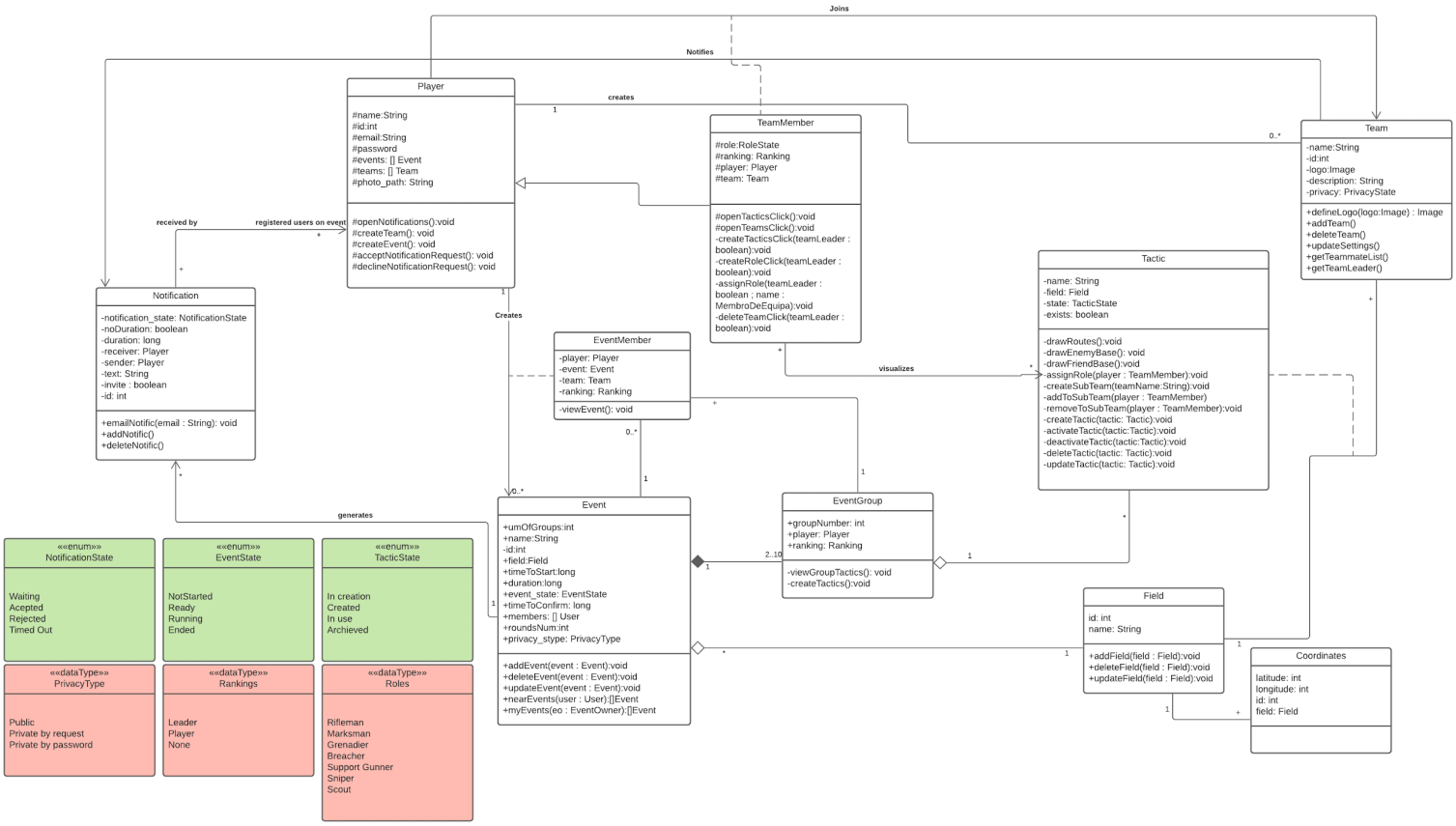
Descrição	O jogador cria um evento, para o criar, define as características do evento
Pré-condições	Jogador autenticado com sucesso Jogador clicou na opção de eventos no menu principal
Cenário Principal	1. O jogador escolhe a opção de criar um evento novo. 2. O jogador desenha um polígono no mapa criando assim um campo. 2.1 O sistema guarda o polígono como sendo o mapa de jogo. 3. O jogador executa o UC15(Alterar definições de evento). 3.1 O jogador executa o UC19(Definir/alterar grupos). 4. O sistema faz uma query à base de dados para confirmar conflitos. 5. O sistema cria o evento. 6. O sistema torna este jogador num ator Dono do Evento. 9. O sistema notifica o Dono do Evento que o evento foi criado com sucesso.
Cenário Alternativo	2.2 O jogador escolhe uma entidade já definida no mapa, sendo esta um campo privado
Pós-Condições	O evento é criado com sucesso. Outros Jogadores podem juntar-se. Outros utilizadores podem ser convidados. O jogador que criou o evento é agora Dono do evento
Cenário de Exceção	6.2.1. A query à base de dados retorna conflito com outro evento previamente existente 6.2.2. O sistema envia notificação com o tipo de conflito.
Pós-Condições	O evento não é criado O Sistema pede um novo preenchimento das definições

UC08 - Criar Táctica

Descrição	O Jogador Líder de equipa cria uma táctica para o mapa selecionado, de forma aos restantes jogadores de equipa poderem visualizar esta táctica e fazerem jogadas coordenadas.
Pré-condições	O jogador tem de membro de uma equipa O jogador tem de ser líder de uma equipa O jogador tem de ter selecionado um mapa O jogador tem de estar no menu da sua equipa

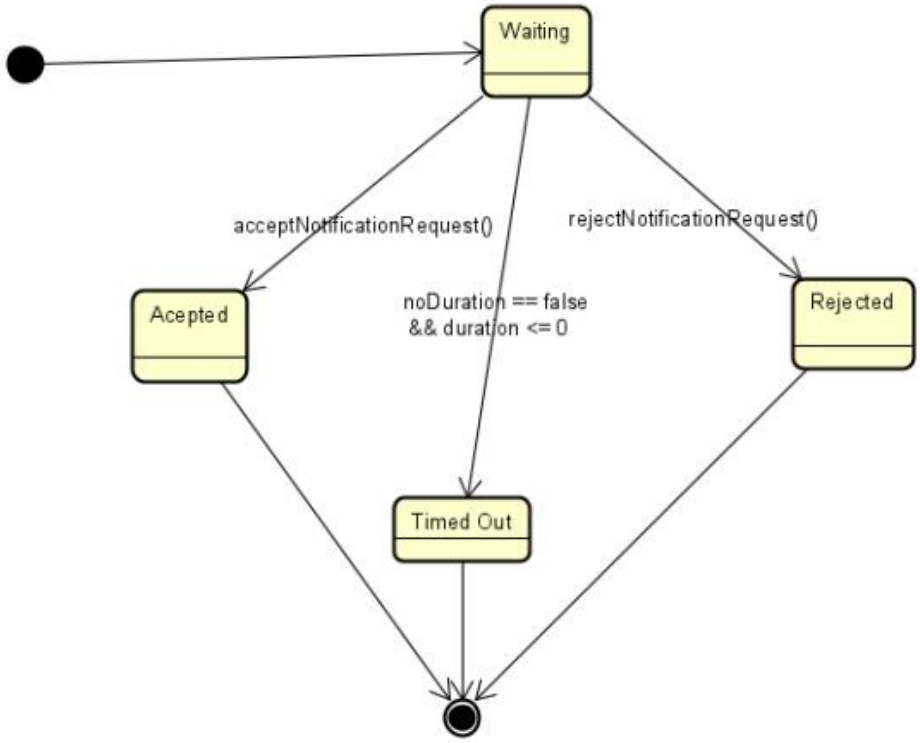
Cenário Principal	1. O Jogador Líder clica na opção de criar nova táctica 2. O Jogador Líder executa UC10(Visualizar representação 2D), ao fazer o upload da imagem do mapa. 3. O Jogador Líder desenha a tática utilizando as diversas ferramentas disponíveis: linha/texto/spray/borracha, alterando o seu tamanho e cor como precisar. 4. O Jogador Líder clica em guardar táctica. 4.1 O sistema guarda a táctica na base de dados.
Cenário Alternativo	N/A
Pós-Condições	A táctica criada fica associada à equipa. A táctica criada fica associada ao mapa. Todos os Jogadores de Equipa podem visualizar a táctica.
Cenário de Exceção	n/a
Pós-Condições	n/a

4 Modelo de Domínio do Sistema TCP

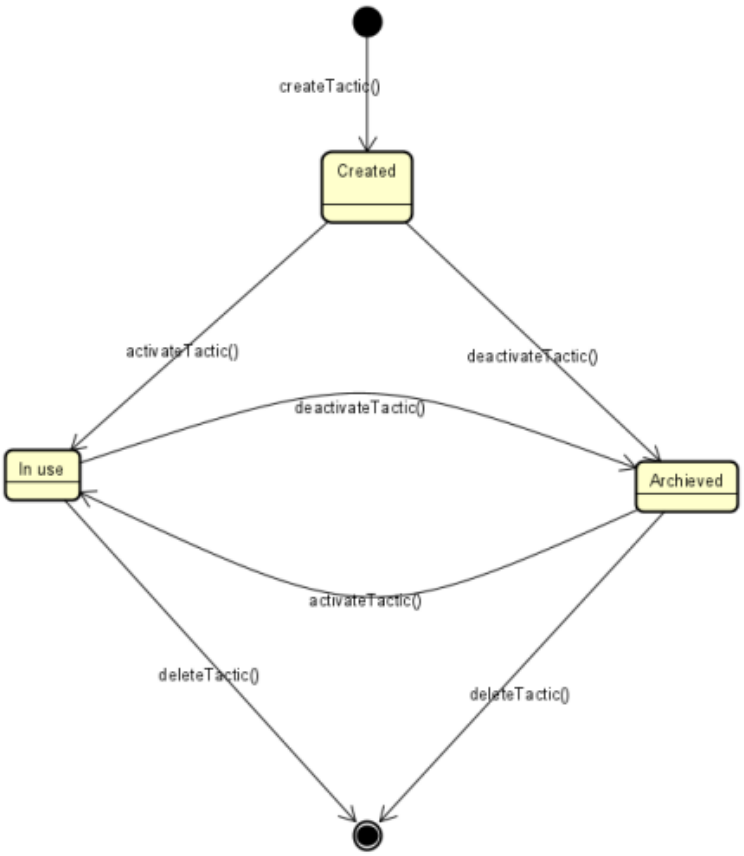


5 Diagrama Máquina de Estados TCP

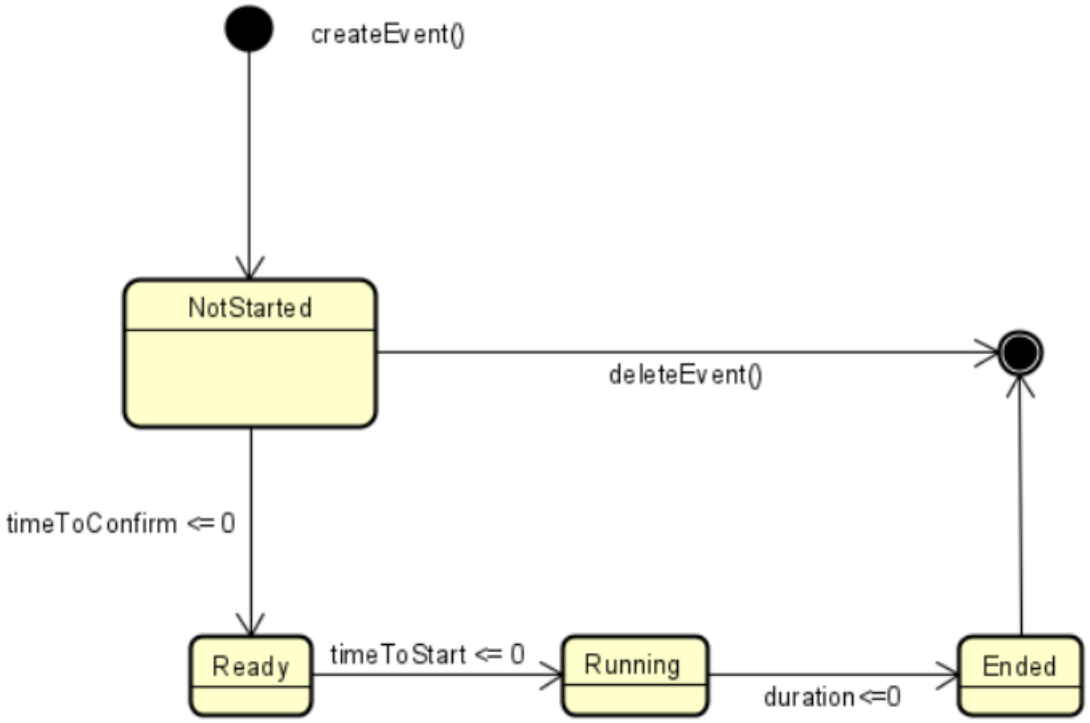
NotificationState Statemachine



TacticState Statemachine



EventState Statemachine



6 Apresentação da Arquitetura do Sistema

O diagrama de blocos representa uma estrutura composicional, na camada de apresentação relacionamos com todas as GUIs que os atores no diagrama de contexto interagem com o nosso SI, na camada de aplicação relacionamos com os nossos modulos/controladores como também APIs externas e uma ligeira referência das bibliotecas externas que usamos. Na camada de Dados, referenciamos como será guardada os dados do sistema, e também quais destes dados interferem com as outras camadas. Também damos uma breve explicação da relação entre os blocos com a utilização de setas, o suficente para perceber as relações sem ficar muito lotado.

"A matriz de CRUD, explicita quais os tipos das acções de processos de negócio que actuam sobre as entidades informacionais. CRUD significa "Create", "Read", "Update" e "Delete":

- "Create" implica a criação de, pelo menos, o identificador dessa entidade.
- "Read" significa que um processo de negócio acede essa entidade.
- "Update" denota uma mudança do estado associado ao identificador dessa entidade.
- "Delete" implica, pelo menos, que o identificador dessa entidade deixa de ser válido, após a execução desta ação a entidade deixa de poder ser manipulada. "[1]

Matriz de CRUD

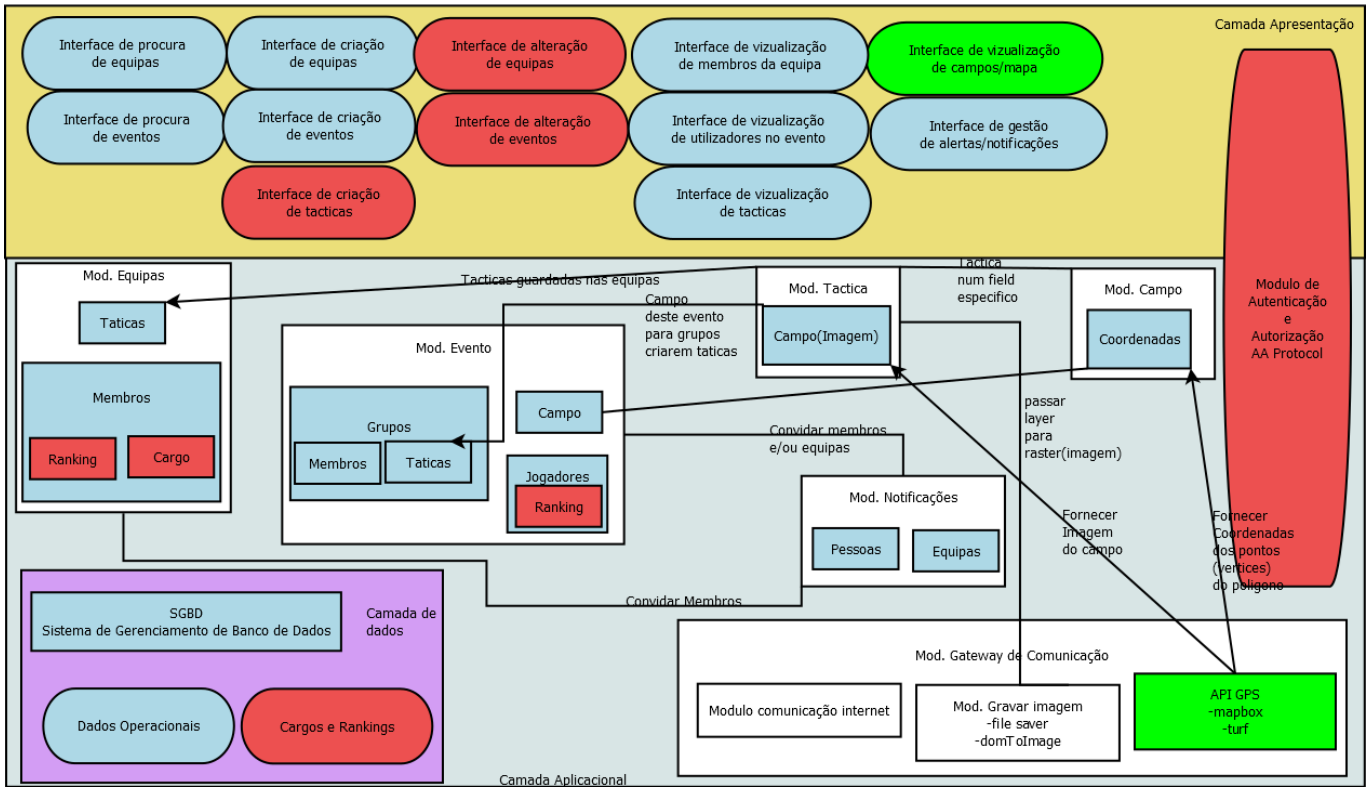
	FR01	FR02	FR03	FR04	FR05	FR06	FR07	FR08	FR09	FR10	FR11	FR12	FR13	FR14	FR15	FR16	FR17	FR18	FR19	FR20	FR21
Jogador	C			R	R	R	RD					CR			R	R		C	C		
Membro de	C		R	R	R	R	RD	R		R		CR			R	R		C	C		R
Líder de Equipa	CRUD	CRUD	CRUD	R	R	CRUD	CRU	CRUD		R		CR			R	R	CUD	C	C		C
Criador de	C			R	R	R	CRU		CRUD			CRUD	CRUD	CRUD	R	R		C	C	C	
Membro de Grupo	C			R	R	R	RD	R	R			CR		R	R	R		C	C	R	
Líder de grupo	C			R	R	R	RD	CRUD	R			CR		R	R	R		C	C	R	
GPS					CU											CU					

No diagrama de blocos temos três camadas, Apresentação, Aplicacional e de Dados. Na camada de apresentação referimos todos os GUIs que estão a ser usados para mostrar como os atores reagem com o SOI no diagrama de contexto. Nesta camada ao utilizar as cores mostramos quais limitações /privilégios são necessárias para a visualização destas interfaces.

Na camada aplicacional mostramos os módulos que usamos para fazer o controlo backend, onde manipulamos os dados para serem mostrados nas interfaces ou para serem guardados na nossa base de dados. Abaixo temos uma breve descrição de cada módulo representado.

- Módulos:
  - Equipa: Neste Módulo fazemos o controlo de várias funcionalidades, como a criação de equipas, adição e remoção de membros, controlo dos rankings e roles dentro desta equipa e também das táticas.
  - Evento: Muito similar ao módulo das equipas, mas dentro de um evento há um campo relacionado a este evento, há grupos(equipas apenas dentro do evento) e estes grupos podem também ter táticas.
  - Tática: Neste módulo apenas controlamos a quem será atribuída a tática e a que campo esta corresponde. Poderá ser atribuída a uma equipa ou a um grupo.
  - Campo: Neste módulo é onde controlamos as coordenadas que são os vértices dos campos desenhamos para os eventos e para as táticas
  - Notificações: por fim neste módulo controlamos as notificações, sejam elas convites ou apenas mensagens, ambos dos casos necessitam de saber qual é a pessoa que envia e a pessoa que recebe, no caso dos convites também é necessário saber para qual equipa/evento é este convite.

Na camada de Dados, referenciamos o que fazemos com os nossos dados, no caso do nosso projeto guardamos toda a informação que está descrita no nosso Modelo de domínio e também referenciamos quais destes dados podem interferir com a camada de apresentação.



7 Especificação de Processos Negócio

Criação de Tática:

Inserir nome tática: O jogador escreve o nome da tática no espaço correspondente.

Escolher equipa/grupo para associar tática: O Jogador clica no drop down de equipas/grupos, que apresenta todas as suas equipas e grupos. O jogador seleciona a equipa ou grupo que pretende associar a tática.

Escolher campo: O Jogador clica no drop down de campos e seleciona o campo que pretende associar a tática.

Carregar imagem localmente XOR Carregar link: O jogador tem duas opções para obter a imagem base para criar a tática de jogo, ou faz o upload através de link ou diretamente do computador.

Desenhar tática:

Dentro deste subprocesso ad-hoc, estão outros 4 processos diferentes, que o Jogador pode reutilizar e executar as vezes que precisas para desenhar a tática.

Selecionar tool: O Jogador escolhe uma das 4 tools: texto/linha/borracha/spray.

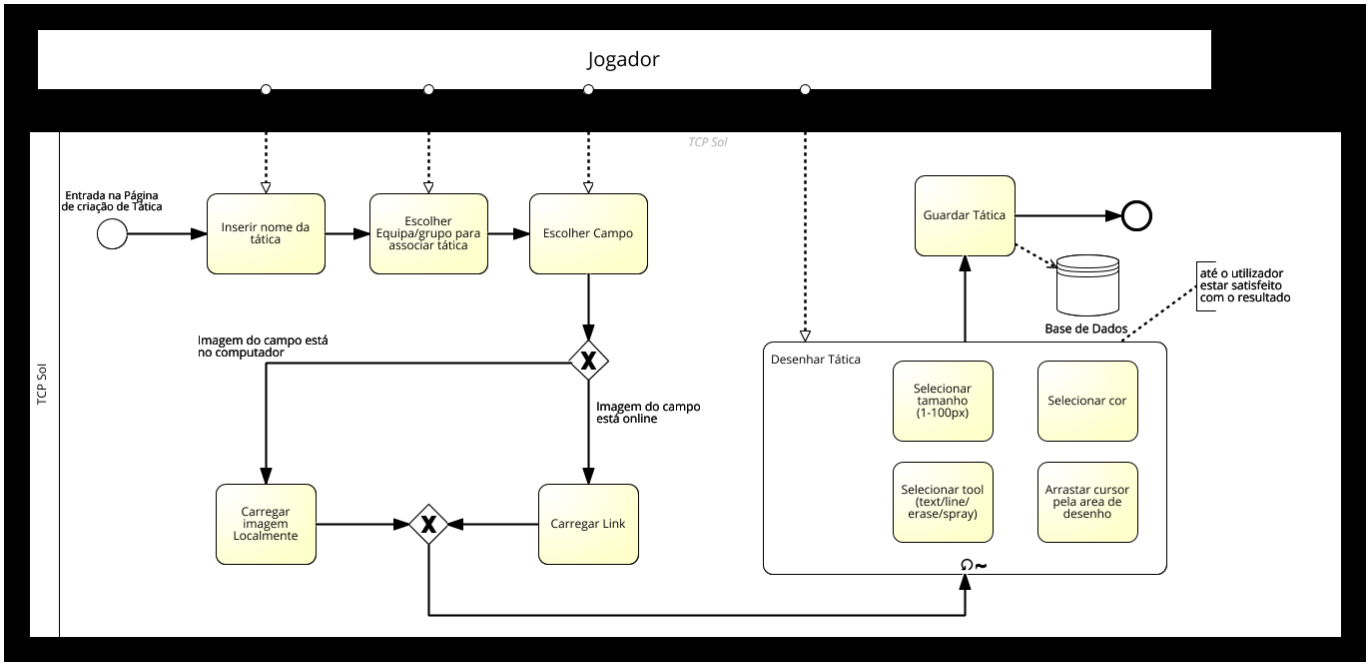
Selecionar cor: O Jogador seleciona a cor que deseja para o seu texto, linha ou spray.

Selecionar tamanho do pincel: O Jogador seleciona o tamanho que deseja para a tool que selecionou

Arrastar cursor pela área de desenho: O jogador clica e mantém premido o botão esquerdo do rato, de forma a desenhar na área de desenho os diferentes elementos da sua tática, tendo como base as tarefas anteriores.

Guardar Tática: O Jogador, quando estiver satisfeito com a sua tática, clica no butão “Create Tactic”, que guarda a tática em associação ao campo e à equipa/grupo que escolheu.

Processo 1: «Criação de Tática»



Criação de Evento:

Sem especificar os subprocessos, o jogador depois de abrir o formulário, faz a tarefa de **Preencher formulário**, podendo ou não **Criar novo campo.** Por fim, faz **Confirmar criação de evento**, onde é guardado o evento, e na se se criar um novo campo, é guardado o campo na base de dados também, terminando o processo de criação de evento.

Subprocesso **Preencher formulário**:

Selecionar campo: O Jogador seleciona o campo do evento, se o campo já estiver criado no mapa, o Jogador passa diretamente para **Preencher campos do formulário**, senão, faz o subprocesso **Criar novo campo**, para depois seguir para **Preencher campos do formulário.**

Preencher campos do formulário: O Jogador insere as características do evento no formulário.

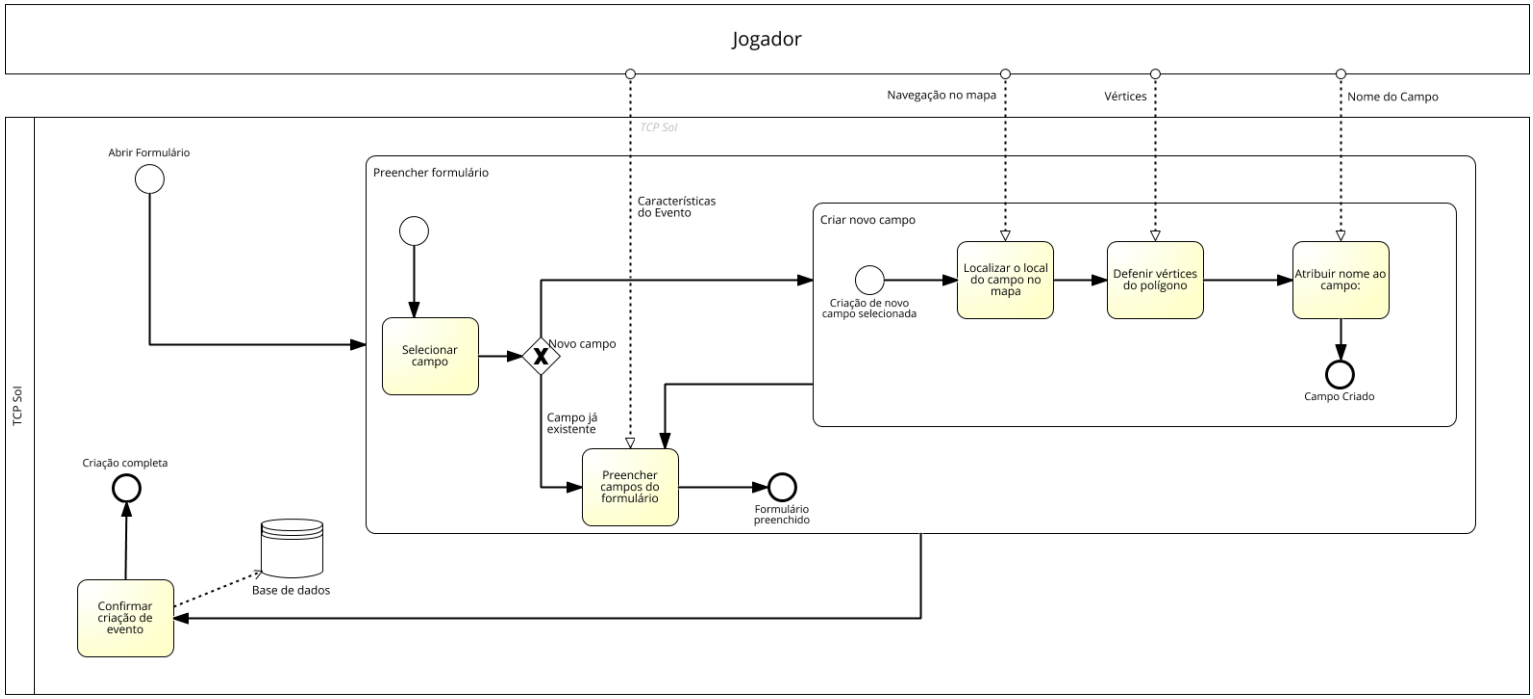
Subprocesso **Criar novo campo:**

Localizar o local do campo no mapa: O Jogador navega no mapa mundo utilizando o rato, de forma a encontrar a zona onde deseja criar o campo.

Definir os vértices do campo no mapa: O Jogador clica no botão de criar polígono, depois pode começar a selecionar os vértices do campo, e para terminar, volta a clicar no vértice inicial.

Atribuir nome ao campo: O Jogador introduz o nome desejado para o campo e continua a criação de evento.

Processo 2: «Criação de Evento»



8 Conclusões e Trabalho Futuro

Para a realização deste projeto assumimos a existência da necessidade de uma plataforma que permitisse à comunidade de airsoft, criar táticas para os seus jogos. Para adicionar mais complexidade ao nosso projeto, assumimos que a comunidade necessitasse também de funcionalidades relacionadas com a organização de eventos e equipas.

Tivémos constrangimentos relacionados com a inexperiência do grupo na área de desenvolvimento web e outros com a limitação das tecnologias definidas para a realização deste projeto, uma vez que desenvolvemos este projeto diretamente em html/css e JavaScript, tornando certas interações de interface que tomamos como garantidas nos websites dos dias de hoje, complexas de implementar.

#	Nome do módulo	Fundamentação
1	Modulo táticas	objetivo do projeto
1	Modulo Campo	Onde conseguimos arranjar a imagem para desenhar as taticas, e onde podemos ver campos e eventos no mapa.
2	modulo equipa	onde podemos juntar um numero de jogadores, controlar os seus cargos e guardar taticas
2	modulo evento	Mesma importancia do modulo anterior, juntar jogadores, organizar por grupos e taticas para cada grupo
3	modulo notificação	Fornecer uma facilitação no processo de convidar para eventos e equipas (nota: um jogador também pode requisitar a entrada numa equipa/evento).

Trabalho Futuro:

- Implementação de uma CDN (content delivery network), de forma a guardar imagens no projeto, guardando os links das imagens na base de dados. Isto melhoraria exponencialmente a utilização da aplicação uma vez que o sistema atual de download e upload de imagens pela parte do utilizador é ineficaz.
- Sistema para procuração de eventos e equipas consoante datas e distâncias definidas pelo utilizador de forma a facilitar esta procura.
- Poder fazer um pedido de junção para equipas eventos. Atualmente apenas é possível juntar a equipas e eventos por convite direto.
- Melhorar a criação de táticas.
- Sistema de Login
- Ser possível criar e editar o perfil do utilizador
- Representação 3D

Referências

[1] P. Alves, A. Vasconcelos , A. Silva. *Arquitecturas de Sistemas de Informação de Referência para Atendimento Multicanal na Administração Pública*, Conference of the Portuguese Association for Information Systems, 2009. [Online]. Available: http://capsi.apsi.pt/index.php/capsi/article/viewFile/102/97

[2] S. Macke, "Dia draws your structured diagrams: Free Windows, Mac OS X and Linux version of the popular open source program", Dia-installer.de, 2021. [Online]. Available: http://dia-installer.de/. [Accessed: 28- Nov- 2020]

[3] *"ClickUp™ | One app to replace them all", Clickup.com, 2021. [Online]. Available: https://clickup.com/. [Accessed: 29- Nov- 2020]*

[4] "Todoist: The to do list to organize work & life", Todoist, 2021. [Online]. Available: https://todoist.com/ . [Accessed: 26- Oct- 2020]

[5] "UsabilityHub | User Testing and Usability Research Platform", Usabilityhub.com, 2021. [Online]. Available: https://usabilityhub.com/ . [Accessed: 13- Dec- 2020]

[6] "Cloud Application Platform | Heroku", Heroku.com, 2021. [Online]. Available: https://www.heroku.com/ . [Accessed: 28- Oct- 2020]

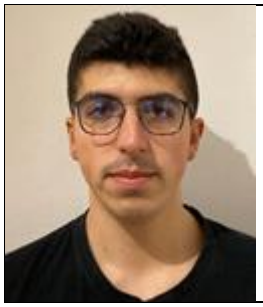
[7] "Online Diagram Software & Visual Solution | Lucidchart", Lucidchart, 2021. [Online]. Available: https://www.lucidchart.com/ . [Accessed: 30- Nov- 2020]

[8] "What is Use Case Diagram?", Visual-paradigm.com, 2021. [Online]. Available: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/. [Accessed: 03- Dec- 2020]

[9] "Signavio | The Only All-in-One Business Process Software", Signavio | The Only All-in-One Business Process Software, 2021. [Online]. Available: https://signavio.com/. [Accessed: 22- Jan- 2021]

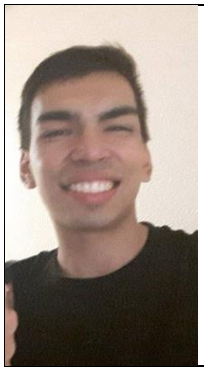
[10] "What is State Machine Diagram?", Visual-paradigm.com, 2021. [Online]. Available: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-state-machine-diagram/. [Accessed: 08- Dec- 2020]

Biografia dos Autores



(50039473) Jaime Ferreira

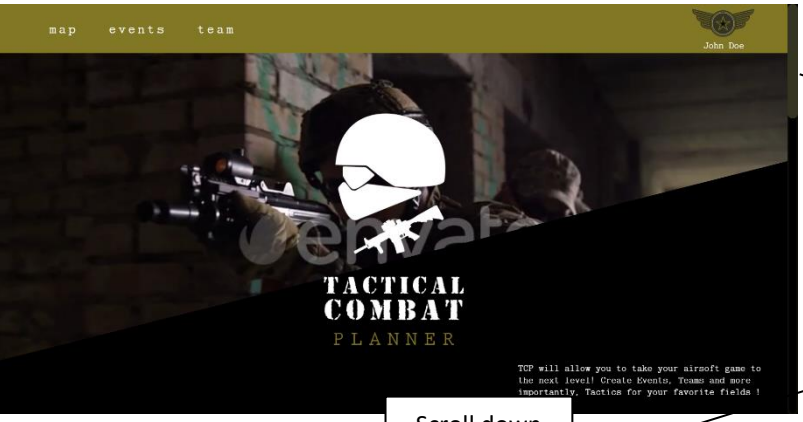
Sou o Jaime, sou aluno do 3º ano de Engenharia Informática da Universidade Europeia, gosto imenso de programação web e mobile, tanto front-end como back-end e tenho como objetivo profissional, gerir uma equipa de programadores, e dessa forma desenvolver projetos.



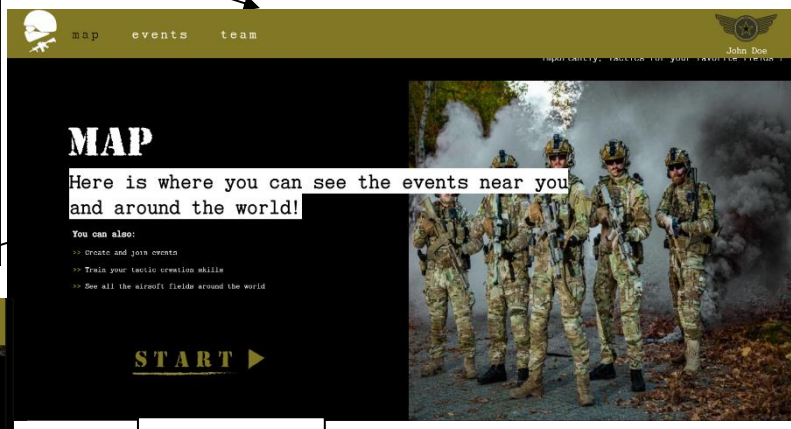
(50039077) Fernando Lamar

Sou o Fernando, também sou aluno do 3º ano de Engenharia Informática da Universidade Europeia, e as áreas da programação que mais me cativam são as de Artificial Intelligence/Machine Learning e de jogos. Pretendo me aventurar nestes dois caminhos e encontrar uma profissão onde eu mais me enquadre.

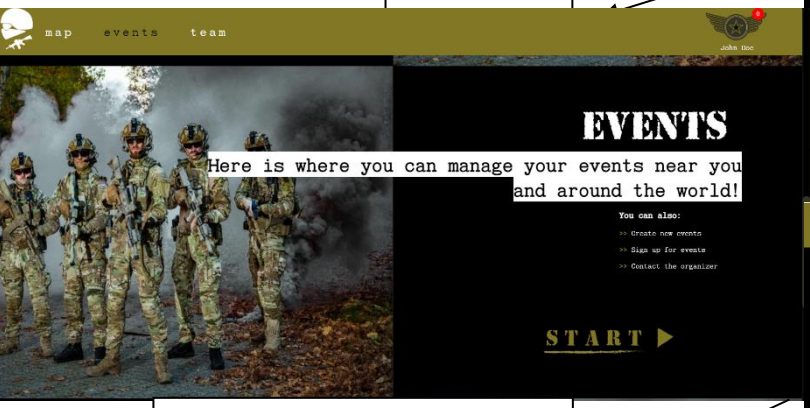
Anexo A: Manual de Utilização da Aplicação TCP



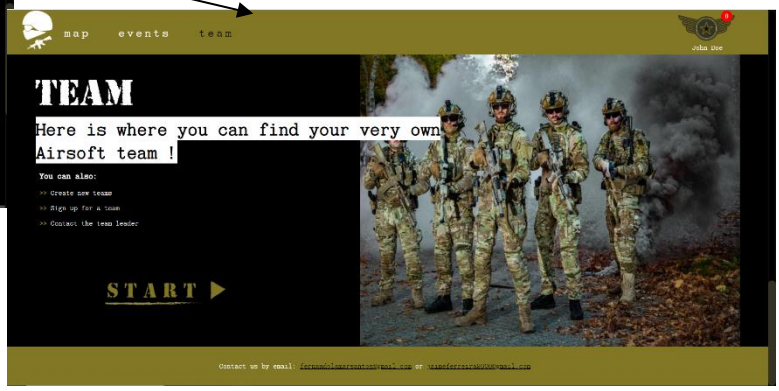
Scroll down



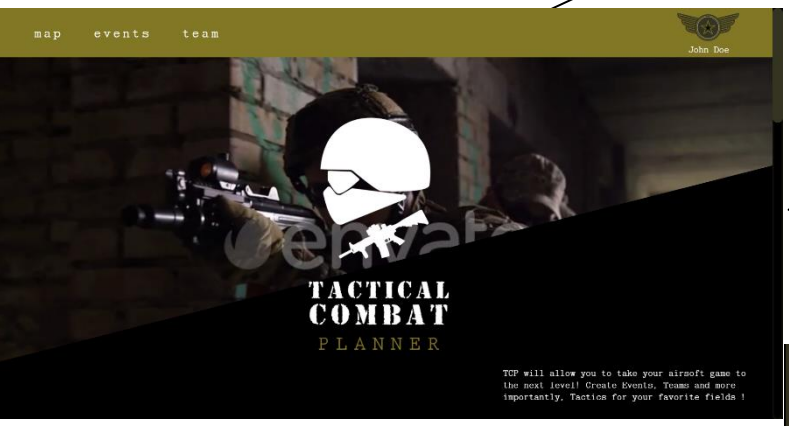
Scroll down



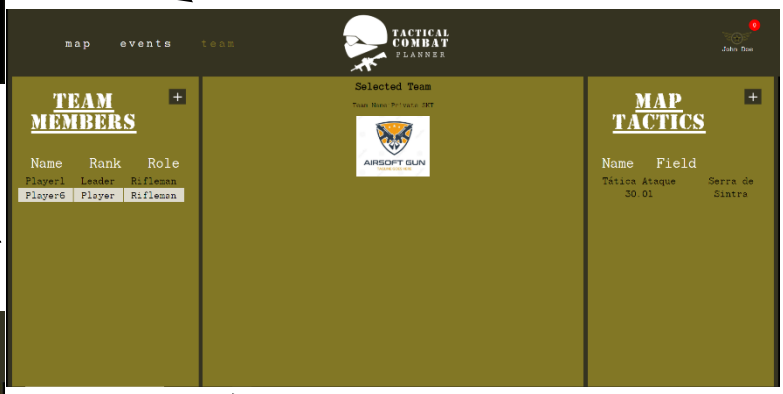
Scroll down



Clique em start ou em “team” na navegação



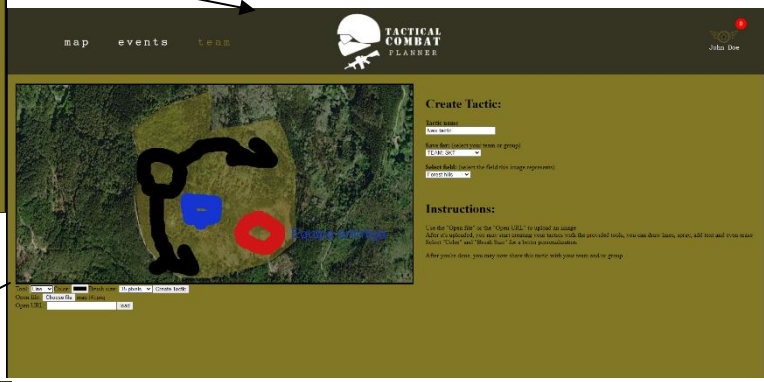
Clique numa equipa específica. Aqui o Jogador pode ver e gerir os membros da sua equipa, adicionar novos, criar novas táticas ou ver as já existentes.



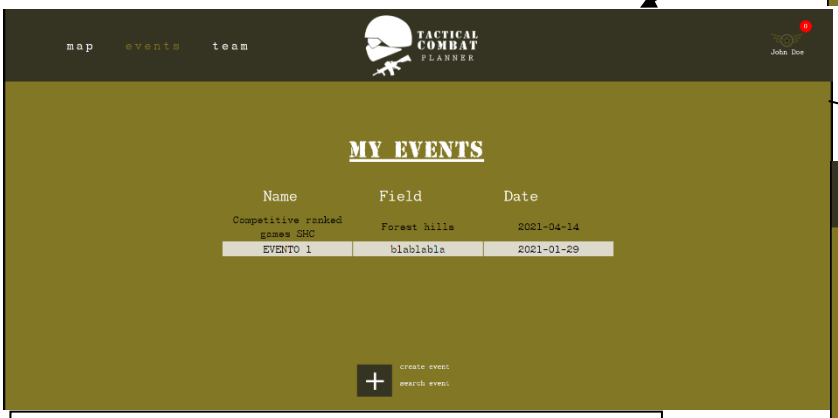
Selecionar uma tática já existente

Clique no “+” em “map tactics” para criar um tática nova

Desenhar tática utilziando as ferramentas fornecidas

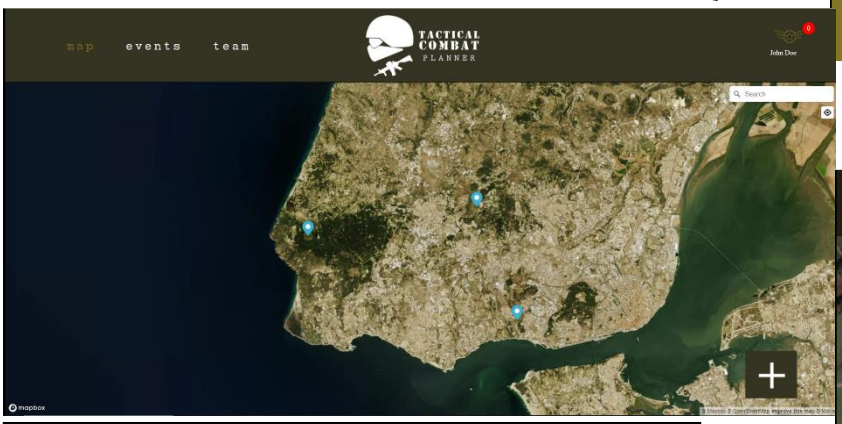


Clicar na tab de eventos

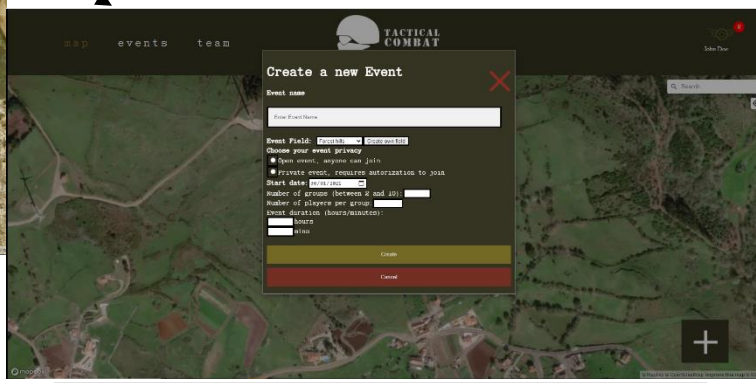


Clicar num evento específico para ver o lobby do evento

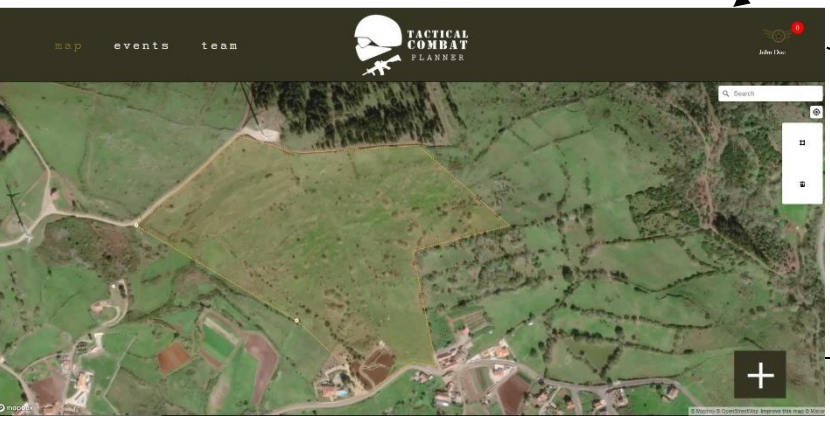
Clicar em “map” na navegação



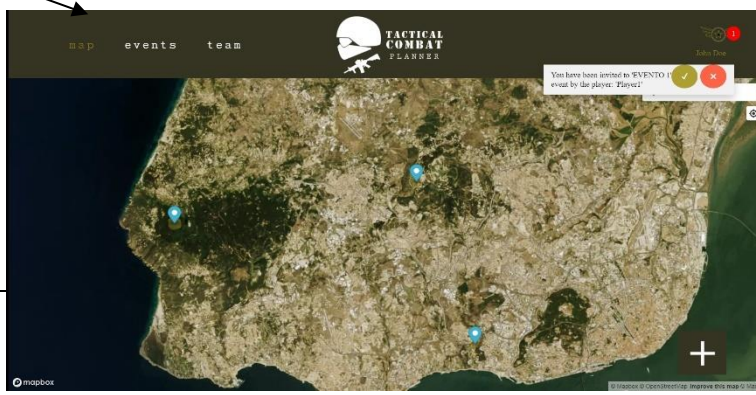
Clicar no “+” no canto inf. esquerdo



Clique para criar novo campo “create new field”, clique no polígono do lado direito, e desenho do polígono.



Clique no nome do Jogador (Notificações)



Link https://tacticalcombatplanner.herokuapp.com/

Anexo B: Frameworks utilizadas no desenvolvimento da solução

- Astah (url: https://astah.net/) usamos para desenvolver o diagrama casos de uso e máquinas de estado
- Lucidchart (url: https://www.lucidchart.com/pages/) usamos para desenvolver o nosso modelo de domínio, razão: a plataforma permite dois colaboradores trabalharem simultaneamente no mesmo diagrama/projeto.
- Signavio (url: https://www.signavio.com/) usamos para desenvolver o diagrama de componentes em notação BPMN
- Dia (url:http://dia-installer.de/) usamos para desenvolver o diagrama de contexto e o diagrama de blocos.
- clickup (url: https://clickup.com/) usamos para definir e planear um cronograma formal de implementação
- todoist (url: https://todoist.com) usamos para definir tarefas informalmente
- usabilityhub (url: https://usabilityhub.com/) usamos para testar a nossa interface
- heroku(url: https://www.heroku.com/) usamos para por a nossa app online

- Camada de Apresentação: IDE: VS code, technologies: HTML / CSS / Node.js
- Camada Lógica de Negócio: IDE: VS code, technologies: Javascript
- Camada de Dados: Plataforma: remote MySQL, technology: MySQL

Anexo C: Levantamento de Requisitos do Sistema TCP

Requisitos Funcionais

#	Nome do Requisito	Descrição	Pri.
FR01	O sistema deverá permitir que os utilizadores criem equipas	Qualquer utilizador deverá ser capaz de criar uma equipa	Alta
FR02	O sistema deverá permitir que os utilizadores convidem outros utilizadores para uma equipa	Após a criação de uma equipa, o utilizador criador poderá convidar outros utilizadores (ou pessoas que não estejam inscritas na app) para a sua equipa.	Média
FR03	O sistema deverá permitir alterações nas definições das equipas	Após a criação de uma equipa, o utilizador criador poderá fazer alterações sobre o mesmo, como por exemplo no nome da equipa.	Baixa
FR04	O sistema deverá permitir que o utilizador consiga procurar por equipas	Deverá haver uma funcionalidade onde os utilizadores poderão procurar por equipas.	Média
FR05	O sistema deverá permitir que os utilizadores acedam a uma representação cartográfica	Deverá existir uma funcionalidade onde o utilizador consiga ver o mapa em 2D e navegar por ele, procurando assim novos campos de jogo ou eventos já criados.	Alta
FR06	O sistema deverá permitir um sistema de hierarquia e títulos dentro das equipas	Dentro de uma equipa o utilizador responsável pela mesma poderá atribuir cargos aos seus membros, facilitando assim o processo de criação de táticas posteriormente.	Média
FR07	O sistema deverá conseguir mandar notificações aos utilizadores	Sempre que houver alguma alteração, por exemplo, num evento que um utilizador esteja inscrito o sistema deverá lhe notificar	Baixa
FR08	O sistema deverá fornecer ferramentas para a criação de táticas	dentro das táticas, na visualização de um campo um líder de equipa poderá desenhar no mapa no âmbito de criar táticas.	Alta
FR09	O sistema deverá permitir que se dividam grupos dentro do evento	Dentro do evento, o utilizador criador deverá poder conseguir atribuir uma pessoa responsável a cada grupo definido.	Média
FR10	O sistema deverá permitir que os utilizadores dentro de uma equipa visualizem táticas.	As equipas poderão ter táticas para campos de jogos individuais e os membros desta equipa deverão poder aceder-lhos.	Média
FR11	O sistema deverá manter registo do utilizador autenticado	Deverá guardar qual utilizador está autenticado para minimizar as comunicações com a base de dados	Alta
FR12	O sistema deverá permitir que os utilizadores criem eventos	Qualquer utilizador deverá ser capaz de criar um evento	Alta
FR13	O sistema deverá permitir que os utilizadores convidem outros utilizadores para um e eventos	Após a criação de um evento, o utilizador criador poderá convidar outros utilizadores (ou pessoas que não estejam inscritas na app) para o seu evento.	Média
FR14	O sistema deverá permitir alterações nas definições dos eventos	Após a criação de um evento, o utilizador criador poderá fazer alterações sobre o mesmo, como por exemplo na data de realização de um evento.	Baixa
FR15	O sistema deverá permitir que o utilizador consiga procurar por eventos	Deverá haver uma funcionalidade onde os utilizadores poderão procurar por eventos	Média
FR16	O sistema deverá permitir que o utilizador consiga procurar por campos de jogo	Deverá haver uma funcionalidade onde os utilizadores poderão procurar por campos de jogo (estes têm de ser previamente estabelecidos como campos oficiais).	Média
FR17	O sistema deverá permitir que o utilizador guarde campos dentro de uma equipa	Deverá permitir que o utilizador crie tácticas para os campos associados com a sua equipa.	Média
FR18	O sistema deverá permitir que o utilizador se junte a uma equipa	Deverá permitir que um utilizador entre numa equipa que não esteja protegida de alguma forma (aberta ao público)	Média
FR19	O sistema deverá permitir que o utilizador se junte a um evento	Deverá permitir que um utilizador entre num evento que não esteja protegido de alguma forma (aberta ao público)	Média

FR20	O sistema deverá enviar notificações aos utilizadores dentro de uma equipa	Deverá notificar os utilizadores inscritos numa equipa sempre que haja alguma alteração na equipa, seja alteração nas definições, alteração nas tácticas ou remoção de tácticas	Baixa
FR21	O sistema deverá enviar notificações aos utilizadores dentro de um evento	Deverá notificar os utilizadores inscritos num evento sempre que haja alguma alteração nas definições do evento, quando este esteja prestes a começar/acabar.	Baixa
FR22	O sistema deverá permitir que o utilizador faça o download da imagem do mapa	Deverá permitir que o utilizador faça o download da imagem do mapa que selecionou ou criou para poder usá-lo como entender	Alta
FR23	O sistema deverá permitir que o utilizador faça upload de imagens	Deverá permitir que o utilizador faça upload de imagens para poder criar tácticas por cima destas imagens	Alta

Requisitos Não Funcionais

#	Nome do Requisito	Descrição	Pri.
NFR01	O sistema deverá ser suportado em plataforma ambiente web(browser).	Irá ser usado o Heroku (Cloud application platform) para suportar a nossa web application	Alta
NFR02	O sistema deverá se comunicar com um servidor base de dados.	A base de dados deverá manter registo de todos os utilizadores, equipas e eventos	Alta
NFR03	O sistema deverá ser desenvolvido na linguagem JavaScript sobre um esqueleto HTML/CSS.	--	Alta
NF04	O sistema deverá ocultar informação pessoal.	O sistema não apresentará aos usuários quaisquer dados de cunho privativo (ex: passwords).	Média

Anexo D: Especificação das Interfaces API

Mapbox (GPS)

Descrição: Esta API tem o objetivo de fornecer ao sistema um mapa do mundo e a localização do utilizador.				Versão: 2.1.0
1	Fornecer Localização		**Descrição:** Este serviço permite localizar o utilizador no mapa.	
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Pedido	**String**	S	**Pedido de localização do utilizador à API**
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	GPS signal	**GPS**	**S**	**Fornece a localização do utilizador ao sistema**
2	Fornecer Mapa		**Descrição:** Este serviço permite ao sistema mostrar ao utilizador um mapa.	
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Zoom	**int**	**S**	**Valor que define a proximidade no mapa**
	Access Token	**String**	**S**	**String que confere ao sistema as permissões necessárias**
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Mapa	**Map**	**S**	**Mapa que o utilizador pode interagir**

Heroku

Descrição: Esta API tem o objetivo de disponibilizar a nossa aplicação (TCP) numa cloud, permitindo aos utilizadores acederem à aplicação através da web.				Versão: 2.6.1
1	Disponibilizar TCP na web	**Descrição:** Este serviço permite que o TCP seja acedido através de um link de internet.		
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Link do github	**String**	S	**Todos os ficheiros relativamente à parte cliente e à parte servidor do projeto, fornecidos ao Heroku através do github.**
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Link de Acesso ao TCP	**String**	**S**	**Fornece o link que os utilizadores podem por num navegador para aceder ao TCP**

Express+Node.js

Descrição: Esta API permite ao SoI comunicar com a base de dados.				Versão: 5.0
1	Permitir Acesso	**Descrição:** Este serviço permite a autenticação do TCP à base de dados		
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	User	**String**	S	**String de User fornecida à base de dados**
	Password	**String**	S	**String correspondente à password fornecida à base de dados**
	Database	**String**	S	**String da base de dados fornecida à base de dados**
	Port	**int**	S	**Num inteiro correspondente ao port fornecido à base de dados**
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Acesso	**Conexão**	**S**	**Fornece a localização do utilizador ao sistema**
2	Fazer Queries	**Descrição:** Este serviço permite ao sistema obter informação vinda da base de dados		
	Parâmetros de Input			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Query	**String**	**S**	**String que é passada à base de dados de forma a obter dados**
	Parâmetros de Output			
	Atributo	Tipo	Obr.	Descrição do Atributo
	Dados	**JSON**	**S**	**Retorna objeto ou objetos no formato JSON**

Anexo E: Funcionamento da Unidade Curricular

Nós concordamos que a unidade curricular foi extremamente benéfica para a nossa formação, uma vez que apreendemos conhecimentos essenciais para um engenheiro informático que, mais que programar, modela o caminho para os programadores. Adquirimos, consolidamos e somamos conhecimento sobre:

- Diagramas de contexto que definem as tecnologias externas, os interfaces necessários para o desenvolvimento do projeto e ainda os atores que interagem com as interfaces.

- Diagramas de blocos que oferecem uma vista geral do projeto, de forma a ajudar alguém que não esteja familiarizado com o projeto a entender como funciona.

- Definir os requisitos funcionais do projeto, de forma a poder fazer um diagrama de casos de utilização, que explicita as funcionalidades do projeto e a interação entre elas mesmas e os diferentes atores; a explicitação correta de cada um dos casos de uso é essencial para uma fácil construção do modelo de domínio e implementação do projeto, uma vez que o que é necessário desenvolver é definido previamente.

- Modelos de Domínio, absolutamente essenciais para a implementação de qualquer projeto.

- Máquinas de Estados que explicitam o funcionamento dos diferentes estados e categorias de estados definidos no modelo de domínio como enumerações.

- As diferenças, as relações e a importância das três camadas na arquitetura de sistemas de informação: Apresentacional, aplicacional e de dados.

Depois de refletir sobre possíveis sugestões para unidade curricular, ambos concordamos que talvez possa ser positivo a realização de mais exercícios práticos. Gostaríamos apenas de realçar que para além do feedback das apresentações das várias versões do Pitch ser extremamente útil, a preocupação demonstrada pelo professor Gabriel Pestana, deu-nos a ambos a motivação para trabalhar e consolidar os conhecimentos lecionados o melhor possível.