



Faculdade de Design,  
Tecnologia e Comunicação  
Universidade Europeia

# Relatório Final – *Better Camp*

## **Etapa Final**

Licenciatura em Engenharia informática

2021/2022

Discentes:

Rafael Santos 20190800

João Alves 20190908

# Índice

1	ENQUADRAMENTO.....	3
2	IDEIA- A <i>BETTER CAMP</i> .....	4
2.1	<i>STAKEHOLDER</i> .....	5
2.2	<i>MAIN TASKS</i> .....	6
2.3	<i>COMPETITIVE ANALYSIS</i> .....	7
3	<i>USER RESEARCH</i> .....	9
3.1	<i>AUDIENCE/TARGET</i> .....	9
3.2	<i>PERSONAS</i> .....	10
4.	<i>ANALISE DE DADOS</i> .....	15
4.1	<i>DIAGRAMA SOI</i> .....	15
4.2	<i>CENARIOS NEGOCIO</i> .....	16
4.3	<i>UML</i> .....	18
5.	<i>PROGRAMACAO WEB</i> .....	19
5.1	CASOS DE UTILIZAÇÃO E GUIÕES .....	19
5.1	DOCUMENTAÇÃO REST.....	25
6	SISTEMAS INFORMAÇÃO GEOGRAFICOS .....	30

# Enquadramento

A solidão é condição, causa e consequência de variados problemas de saúde física e mental em Portugal e no mundo, como a obesidade, depressão e sedentarismo, podendo afetar qualquer pessoa, independentemente do seu género, idade, rendimento ou classe social. Por sua vez, o isolamento social obrigatório a que a sociedade foi submetida nos últimos anos, no âmbito do COVID-19, veio agravar esta problemática, e mais do que nunca, é necessário criar medidas de combate e prevenção da mesma.

O objetivo deste projeto é precisamente prevenir e combater, com o auxílio das Tecnologias de Informação e Comunicação (TIC), a solidão e exclusão social e, da mesma forma, promover a saúde física e mental e a cooperação e participação/ envolvimento cívico dos cidadãos. De modo a alcançar este objetivo, serão adotadas medidas específicas que envolvam os membros de uma comunidade (online) a promoverem eventos de convívio presencial. O *outcome* desejado é alertar para o perigo de incidência de doenças consequentes da solidão e sedentarismo, como as doenças cardiovasculares, hipertensão e diabetes (as principais causas da mortalidade em Portugal), e da mesma forma, alertar e combater doenças mentais como a depressão e ansiedade, que podem resultar/ conduzir ao suicídio.

A verdade é que a Revolução Tecnológica e o atual paradigma das redes sociais tornaram possível a comunicação e criação de relações entre pessoas de toda a parte do mundo, mas não podemos deixar de refletir sobre algumas consequências do uso excessivo desta mesma tecnologia, como a deterioração de relações sociais e o aumento da solidão e sedentarismo. Nesta era digital, nunca estivemos tão conectados e tão sós ao mesmo tempo. Como cidadãos do mundo, cabe-nos a todos pensar e agir coletivamente para combater esta pandemia silenciosa.

Assim, o nosso projeto visa a organização de um programa que reúna pessoas de diversas faixas etárias – jovens adultos (18–27 anos), adultos (28–54 anos) e idosos (+55 anos). Este consistirá na organização de atividades inseridas no âmbito de campos de férias, situados em vários pontos do país, em parques naturais, reservas etc..

Este programa visa sensibilizar as pessoas sobre a problemática em questão e promover o envolvimento e a coesão social das diversas faixas etárias, através da realização de atividades e *workshops* que promovem a partilha de informação e cooperação entre os participantes. O foco será em promover hábitos de vida mais saudáveis focados na saúde mental e física, com a coordenação por profissionais destas áreas em questão.

## **Ideia - *A Better Camp***

*A Better Camp*, foi uma ideia que surgiu depois da necessidade de combater alguns problemas que sempre existiram relacionados com cuidados de saúde primários principalmente sedentarismo, obesidade, solidão, doenças estas que são as mesmas que causam mais casos de morbilidade e mortalidade em Portugal.

Um dos grandes objetivos é tornar esta experiência gratuita para toda a gente que poderia participar com alimentação, atividades, transporte e alojamento durante uma semana, e cada uma destas semanas irá se focar em vários grupos etários diferentes, visto que cada um destes grupos necessita de diferentes intervenções.

Este programa visa sensibilizar as pessoas sobre a problemática em questão e promover o envolvimento e a coesão social das diversas faixas etárias, através da realização de atividades e *workshops* que promovem a partilha de informação e cooperação entre os participantes. O foco será em promover hábitos de vida mais saudáveis focados na saúde mental e física, com a coordenação por profissionais destas áreas em questão.

Estes campos vão ser situados em campos, reservas ou parques naturais com objetivo também de sensibilizar para limpeza destes espaços.

## ***Stakeholder***

Uma possível entidade que poderia estar interessada em participar ou ajudar no projeto seria a Administração Regional de Saúde de Lisboa e Vale do Tejo, Instituto Público (**ARSLVT, I.P.**).

Esta entidade tem como missão, garantir à população, da Região de Lisboa e Vale do Tejo, o acesso à prestação de cuidados de saúde de qualidade, adequando os recursos disponíveis às necessidades em saúde e cumprir e fazer cumprir o Plano Nacional de Saúde.

O objetivo dos cuidados de Saúde Primários são a promoção da saúde e prevenção da doença e assim sendo através deste projeto poderá prevenir doenças tais como a obesidade, HTA, doenças cardiovasculares e depressão levando a uma diminuição de morbilidade e mortalidade em Portugal.

Com tudo isto dito, seria uma entidade bastante interessante e que poderia ser algo bastante bom para promover o projeto e também ao mesmo tempo financiar o mesmo, visto que a finalidade do nosso projeto e desta entidade são bastante idênticas.



Fig.1 – Logo ARSLVT, IP

## ***Main Tasks***

O nosso projeto/site visa responder a alguns objetivos neste caso pretendemos que os nossos clientes tenham facilidade em juntar-se a esta experiência. Com isto dito uma das principais tarefas será mostrar informação sobre o campo, o objetivo do projeto e muitas mais informações.

Também procuramos em oferecer experiência e trabalho renumerado a pessoas interessadas a tornarem-se monitores e teremos uma página que irá mostrar benefícios e a missão do projeto, como também a mesma conta com um formulário para possível inscrição para algum tipo de individuo se tornar um monitor. (*Gamification*)

Com o trabalho de pesquisa relativamente á nossa competição conseguimos reparar que seria importante mostramos fotos das atividades e do campo em si, para que futuros clientes tenham uma perspetiva mais clara do mesmo.

Como isto trata-se de um site informativo e não de uma App, achamos interessante visto que não existe a facilidade de existir um rating próprio da mesma, contarmos com uma secção com a opinião de vários clientes e próprias testemunhas.

Por último será interessante mostrar todo o tipo de *workshops* e atividades que vão decorrer no campo, quais os profissionais encarregues pelas mesmas.

## ***Competitive Analysis***

Depois de uma pesquisa detalhada, achamos alguma dificuldade em comparar este projeto com algum tipo de produto no mercado, por isso decidimos recolher alguns campos de férias com diversos temas.

Conseguimos juntar três exemplos de projetos/produtos que ofereciam serviços de certa forma parecido ao nosso projeto de certa forma foram eles o *MyCamp*, Campo Aventura e o *Science4you*. Em cada um destes exemplos vamos mostrar alguns pontos positivos e negativos de cada um destes projetos.

- ***MyCamp***

- **Vantagens:** Site bastante completo e muito intuitivo com muita informação sobre o campo, apresenta bastantes vídeos das atividades e de passadas experiências e conta também com feedback de antigos clientes. Também contam com bastantes prémios e programas escolares e inscrição online.
- **Desvantagens:** Apesar de falarem do campo em questão não mostram nenhum tipo de utilização de sistemas geográficos o que poderia ser uma mais-valia para um cliente ver o campo em primeira mão *online*.

- ***Science4you***

- **Vantagens:** Site simples e bastante apelativo e uma ótima forma de cativar amantes de ciência visto que não existem campos de férias focados nesse ramo.
- **Desvantagens:** O foco desta entidade não é inteiramente os campos de férias, mas sim as lojas com diversos produtos orientados para o tema das Ciências o que faz com que para um utilizador que queira saber mais sobre o campo tenha uma dificuldade extra.  
Não apresenta muitas imagens de antigas atividades e apenas para além de uma breve introdução e explicação apenas apresenta o preço por dia ou por semana e o seu contacto por mail ou telemóvel.

- **Campo Aventura**

- **Vantagens:** Site bastante informativo também com bastantes imagens e testemunhos e com um sistema de comentários de antigos clientes com resposta aos mesmos. Para além do público alvo ser crianças e jovens, procuram realmente assegurar os pais com vários guias para os mesmos e apresentam bastantes vídeos do campo.
- **Desvantagens:** Apesar de muito informativo não parece ser o UI mais apelativo e apresenta um aspeto muito antigo e conta com algumas falhas em termos de interface.



# *User Research*

## **Audience/Target**

O publico alvo para o nosso projeto como já antes mencionado são todas as pessoas entre as idades dos 18 aos 65 e residentes na zona de Lisboa e Vale do Tejo, que pretendam participar e socializar durante uma semana com pessoas da sua faixa etária entre várias atividades e *workshops*.

Será também fácil de presumir que existirão alguns obstáculos, nomeadamente as pessoas mais jovens a trabalhar ou a estudar ou até mesmo a faixa etária mais adulta em relação ao trabalho. Presumimos também que estas pessoas já estariam interessadas e motivadas depois de aconselhamento médico nos Centros de Saúde, sendo também este um dos aspetos importantes da entidade promotora já antes referida.

Um dos aspetos importantes deste projeto é também o facto que o mesmo seria financiado e que o custo não seria algo importante para os intervenientes. Salientar o facto que mesmo que o publico alvo seja este tipo de indivíduos, a publicidade ao projeto tornava-se também importante para pessoas como pais e filhos que gostariam e até aconselhassem a membros da sua família a participar.

## Personas

Depois de perceber um pouco sobre o nosso publico alvo, é necessário nos metermos mais na pele de um possível cliente e pensar que dentro deste estereótipo, que tipo de pessoa iria gostar de conhecer mais sobre o nosso projeto ou até participar no mesmo.

Para isso construímos estas quatro diferentes personas, de vários géneros, idades, estaturas e diferentes objetivos na vida, mas que de diferentes maneiras poderão achar o nosso projeto interessante ou como já dito participar no mesmo.

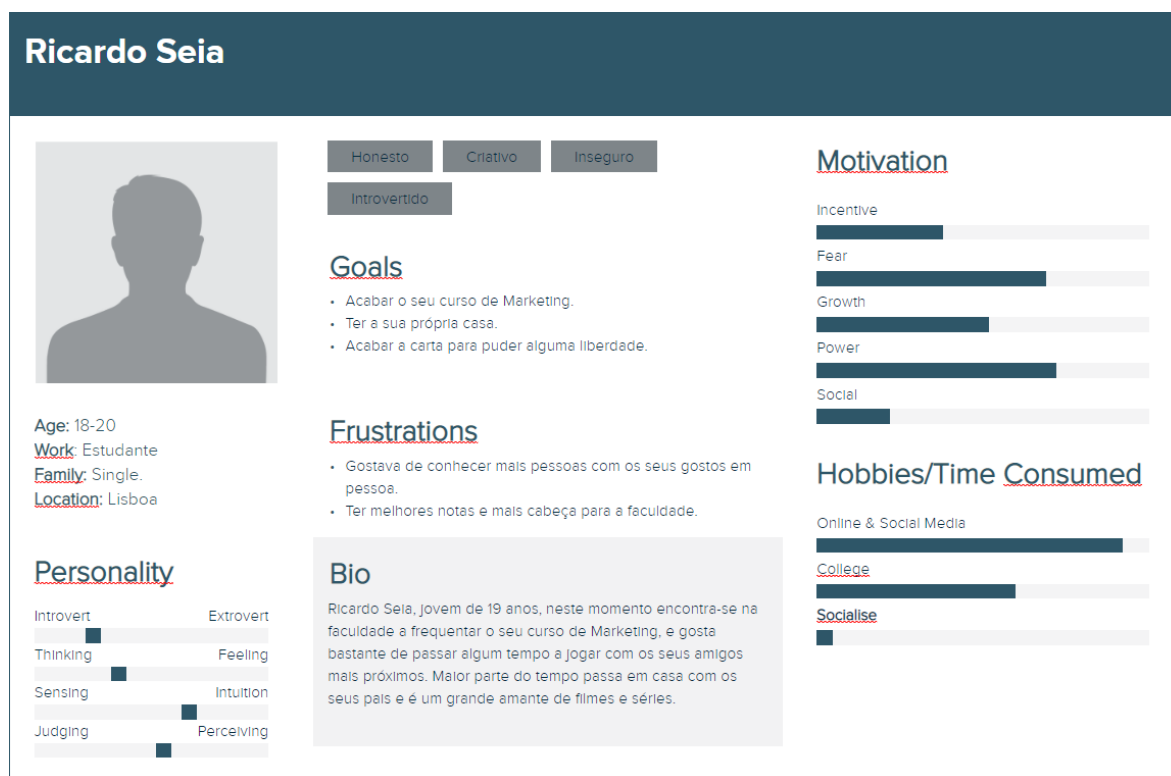


Fig.2 – Persona (Ricardo Seia).

Esta *persona*, trata-se de um jovem bastante novo que ingressou pela primeira vez na faculdade, tem alguns hábitos sedentários, mas que socializa com os seus amigos mais próximos, mas gostaria de alcançar mais pessoas com os mesmos interesses, neste caso pessoalmente.



Fig.3 – UX Journey (Ricardo Seia).



Fig.4 – Persona (Adelaide Portugal).

A nossa segunda *persona*, é uma idosa bastante independente e que gosta bastante de se sentir jovem e de se desafiar no dia-a-dia, tem uma família grande e gosta bastante dos seus netos e gostaria de ser mais ativa e de arranjar um novo hobby visto que sente que tem bastante tempo livre.



Fig.5 – UX Journey (Adelaide Portugal).

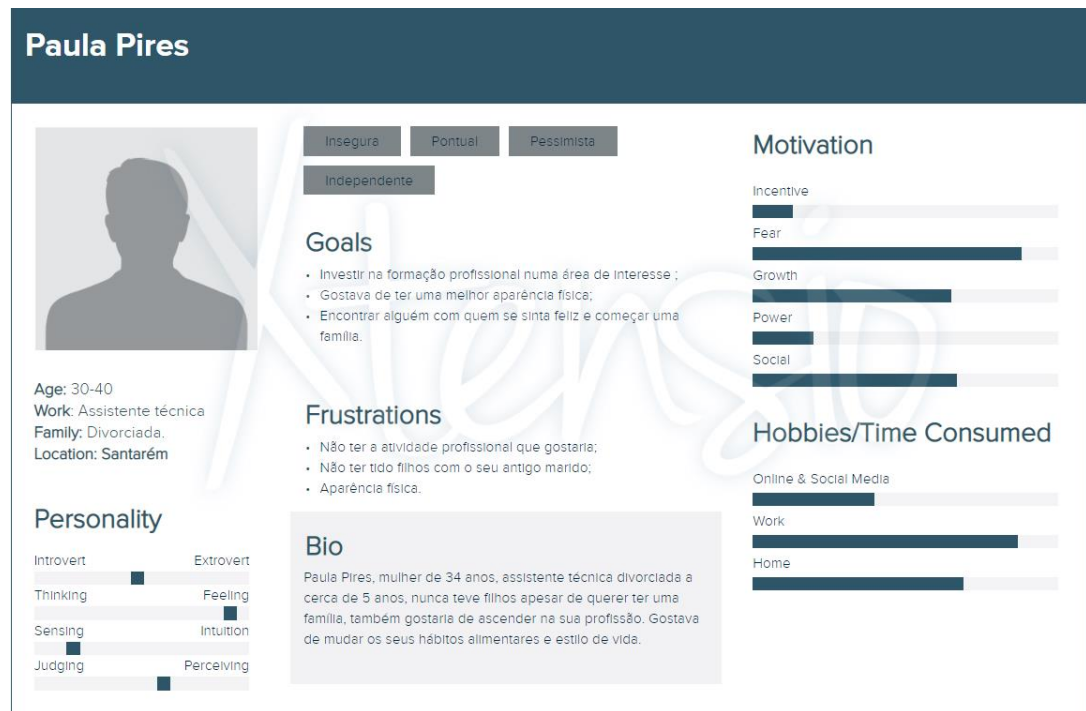


Fig.6 – Persona (Paula Pires).

Decidimos escolher como terceira *persona*, alguém mais adulto empregado, mas que se sente triste com a sua carreira e com a sua aparência e estilo de vida. Esta “mulher” teve um casamento difícil e neste momento é divorciada e gostaria de dar uma reviravolta por completo na sua vida.



Fig.7 – Persona (Paula Pires).

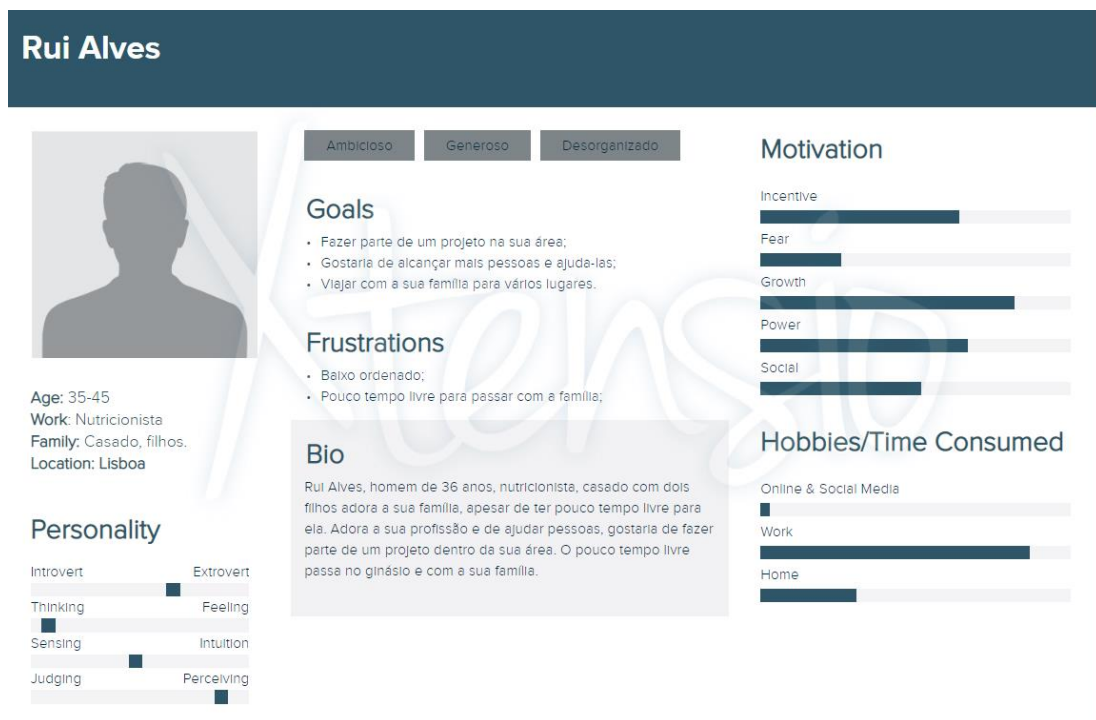


Fig.8 – Persona (Rui Alves).

Finalmente decidimos por último mostrar alguém apaixonado pelo seu trabalho, adulto com família, bastante ambicioso e feliz. Sendo nutricionista gostaria de participar em projetos na sua área e procurava ajudar ainda mais pessoas.



Fig.9 – UX Journey (Rui Alves).

# Análise de Dados

## Diagrama de Contexto

O nosso diagrama de contexto representa todas as roles que interagem com o nosso SoI (*Better Camp*). O nosso sistema conta com vários atores entre eles o utilizador, administrador e monitor e um API o *Leaflet* para criação e utilização e informação geográfica como também criação de rotas como conseguimos ver no diagrama.

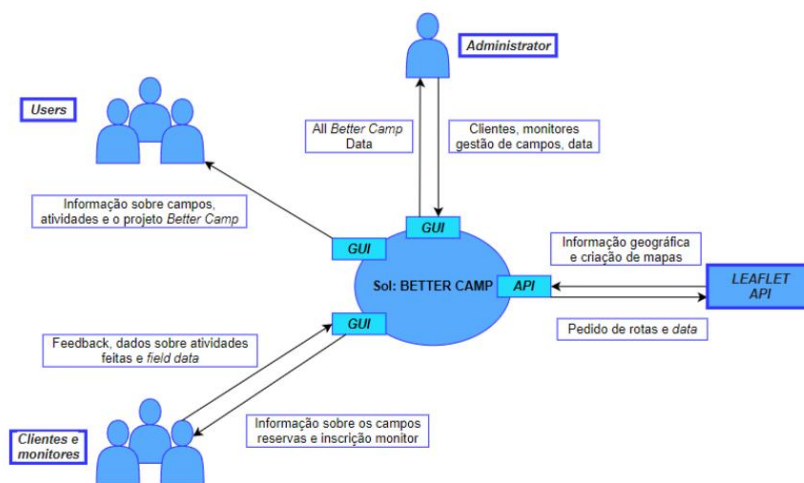


Fig.10 – Diagrama de Contexto (*Better Camp*)

# Cenários de Negócio e BPMN

## 1. Reserva dentro de um campo feita pelo cliente

- **Assunção:** cliente com log-in feito, na página de reservas.
- **Descrição:** o cliente vai ser redirecionado para uma página de reservas em que o mesmo terá que fornecer que campo o mesmo deseja escolher. Depois da escolha do campo, será mostrado todas as semanas disponíveis ainda com lotação disponível.
- O sistema vai receber os dados desta reserva, e vai guardar esta nova reserva como também atualizar a lotação atual do campo e da semana escolhida pelo mesmo.

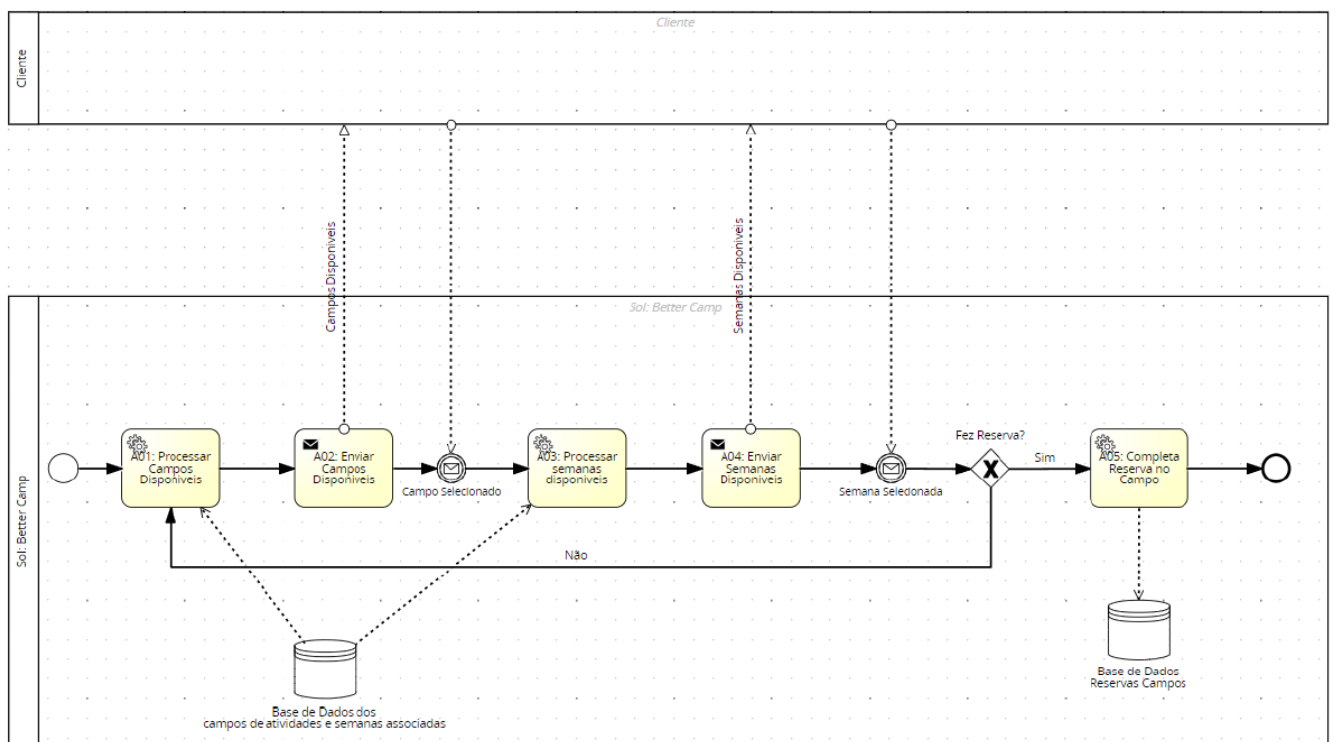


Fig.11 – Diagrama de Colaboração (Cenário 1)



## 2. Administrador aceita pedido de cliente para se tornar monitor

- **Assunção:** admin com log-in feito, na página de pedidos.
- **Descrição:** o *admin* vai ser redirecionado para uma página de pedidos que mostra os vários pedidos feitos pelos clientes, estes pedidos duram apenas 7 dias, estes pedidos podem ser aceites ou recusados, se aceites o cliente torna-se num monitor.
- O sistema vai atualizar o pedido e atualizar o estado do cliente tornando o mesmo num monitor. Se recusado o sistema apenas atualiza o pedido e não o estado do cliente.

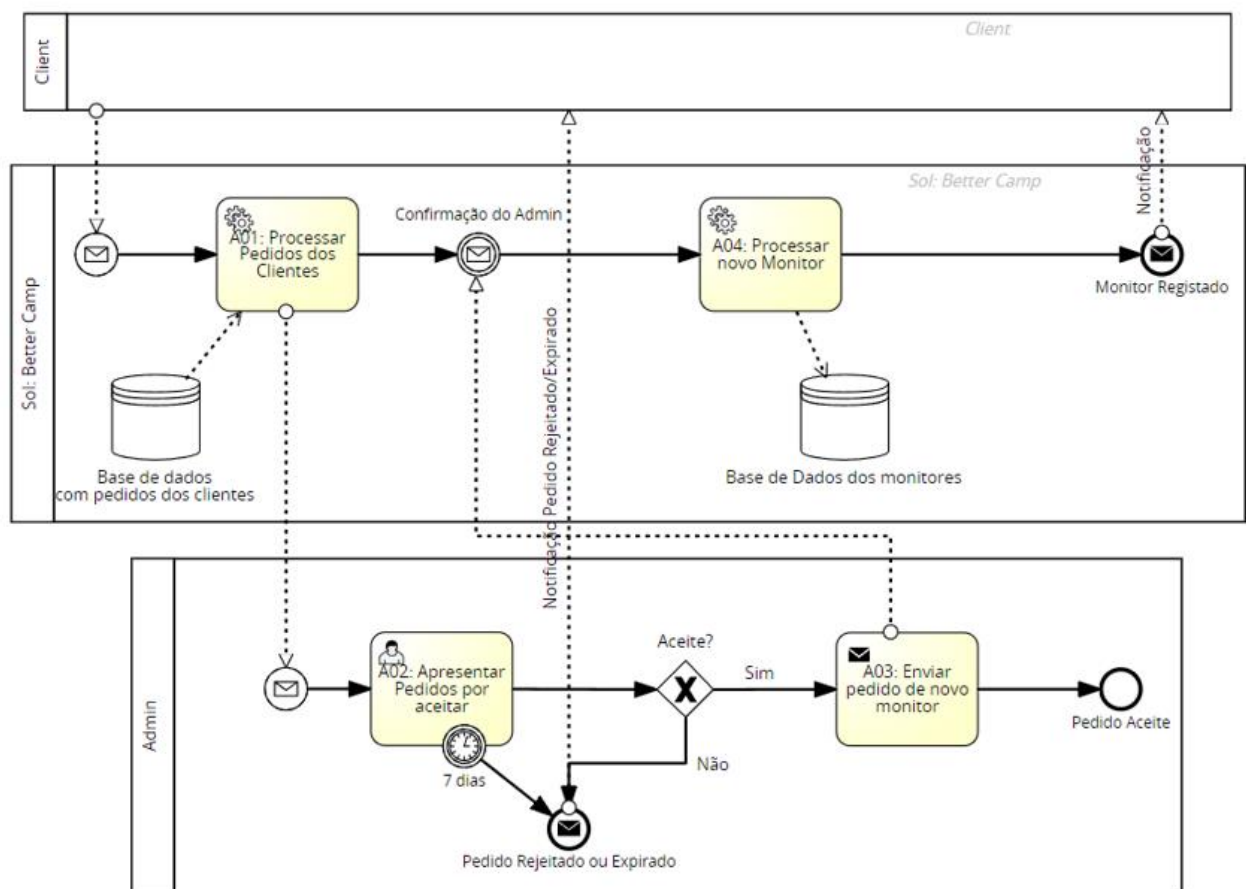


Fig.12 – Diagrama de Colaboração (Cenário 2)

### 3. Administrador associa monitor a campo

- **Assunção:** admin com log-in feito, na página de pedidos.
- **Descrição:** o *admin* vai ser redirecionado para uma página onde poderá selecionar quais os monitores a escolher para x semana. O mesmo terá que preencher em que semana o monitor irá participar.
- O sistema vai adicionar o monitor num campo em uma semana à escolha pelo administrador.

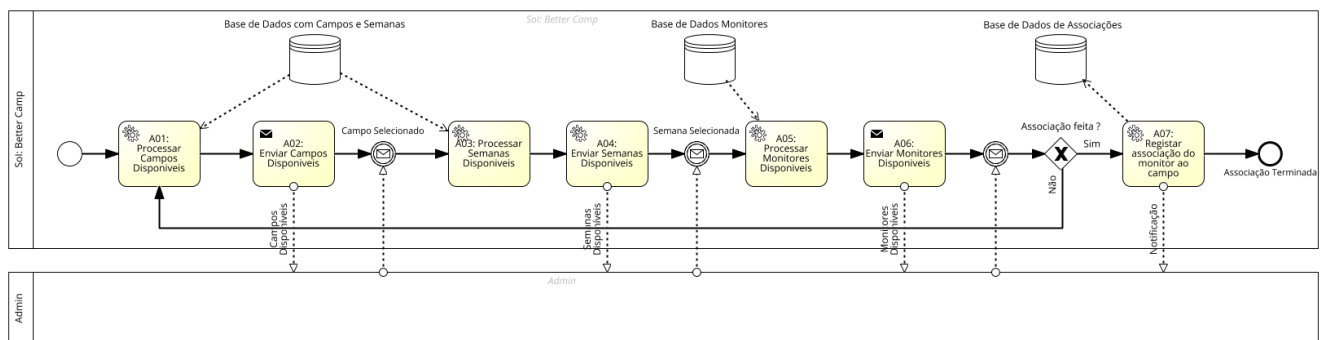


Fig.13 – Diagrama de Colaboração (Cenário 3)

## UML

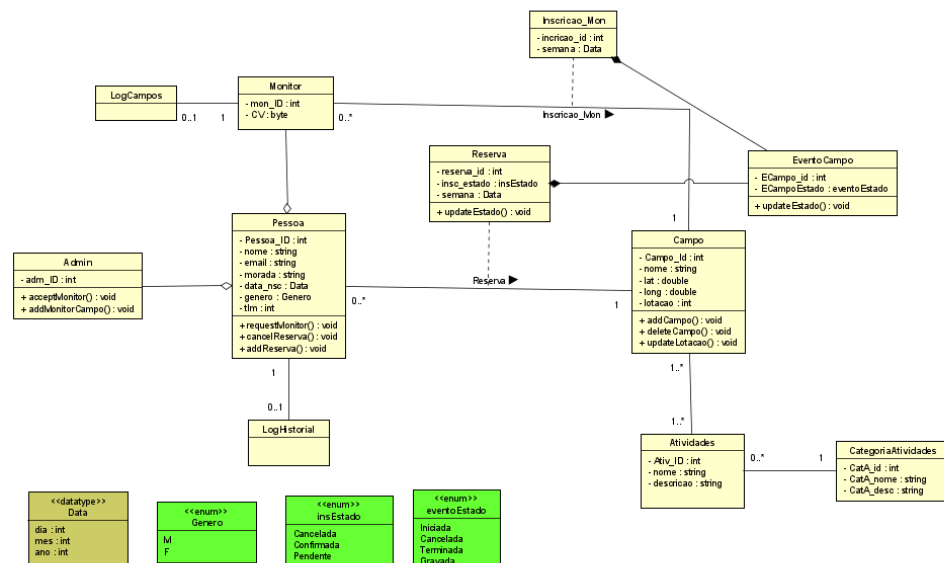


Fig.14 – Diagrama Modelo de Domínio.

# Programação Web

## Casos de Utilização e Guiões

### 1. Reserva de um cliente num Campo:

#### Pré-requisitos:

- a. Registo feito por parte do utilizador;
- b. Login feito da conta do ponto a);

#### Cenário Principal:

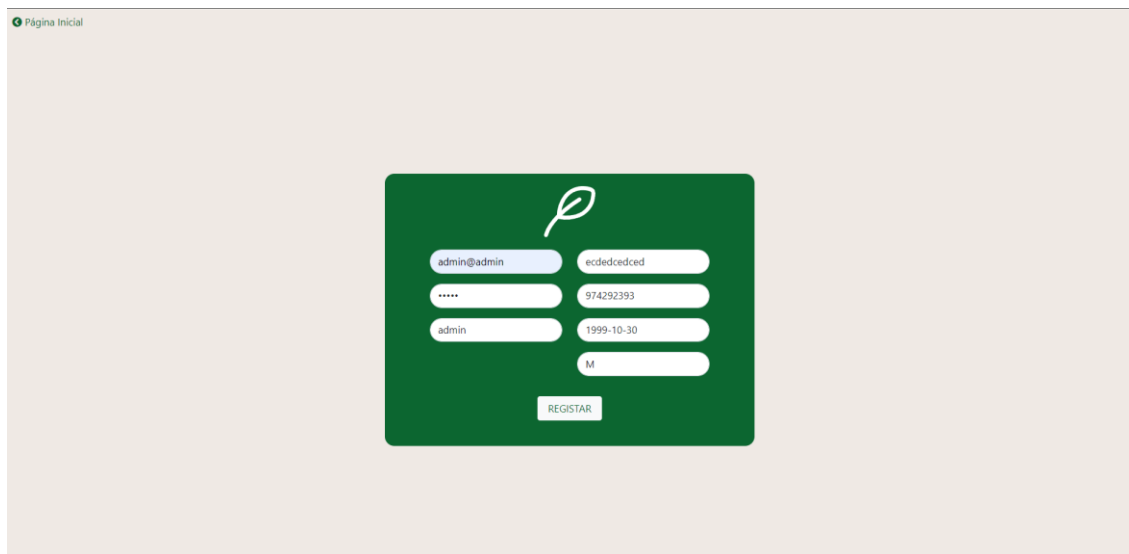
1. Na homepage aceder á tab de “Campos”;
2. Na página de campos, escolher um dos campos disponíveis;
3. Na página do campo escolhido do ponto 2, escolher a semana disponível na *dropbox* apresentada;
4. Finalizar a reserva carregando no botão “Inscrever”.

#### Pós-condições:

- a. Registo da nova reserva adicionado na base de dados (POST);

### Guião do Cenário 1:

#### a. Registo feito por parte do utilizador;



The screenshot shows a registration form titled 'Página Inicial' in the top left corner. The form is set against a dark green background with a white leaf icon at the top. It includes the following fields: an email field with 'admin@admin', a password field with 'echedcedced', a phone number field with '974292393', a name field with 'admin', and a date of birth field with '1999-10-30' and a dropdown menu showing 'M'. A white 'REGISTAR' button is located at the bottom of the form.

Fig.15 – Registo Utilizador (Cenário 1).

**b. Login feito da conta do ponto anterior;**



Fig.16– Login utilizador (Cenário 1).

**1. Homepage aceder “Campos”;**



Fig.17– Página Campos (Cenário 1).

**2. Escolha de campo.**

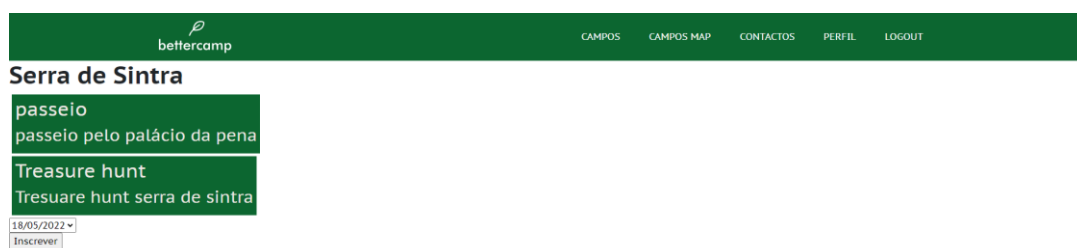


Fig.18– Escolha de Campo “Serra de Sintra” (Cenário 1).

## **2. Pedido de Cliente para tornar-se monitor/*Admin* aceitar pedido/Atribuir monitor a campo:**

### **Pré-requisitos:**

#### **1º parte**

- a. Login feito com uma conta de cliente (não monitor);

#### **2º e 3º parte**

- b. Login feito com uma conta de *admin*;

### **Cenário Principal:**

#### **1º parte**

- 1. Na página de perfil depois do login (cliente), clicar no botão “Pedir para ser Monitor”;

#### **2º parte**

- 2. Na página de perfil depois do login (*admin*), clicar no botão “Ver Pedidos de Monitor”;
- 3. Na página dos pedidos é mostrado os vários nomes dos clientes que fizeram pedidos;
- 4. Aceitar um dos pedidos no botão “Aceitar o monitor”.

#### **3º parte**

- 5. Na página de perfil depois do login (*admin*), clicar no botão “Gerir Monitores”;
- 6. Na página de gestão de monitores é mostrado as datas e campos disponíveis e também os monitores disponíveis;
- 7. Depois de fornecer os dados todos para atribuir um monitor a um campo em x semana, clicar no botão “Submeter”.

### **Pós-condições:**

#### **1º parte**

- a. Registo do novo pedido para monitor adicionado na tabela “monitor\_insc” no campo CV=0 (POST);

#### **2º parte**

- b. *Update* feito na base de dados na tabela “monitor\_insc” no campo CV=1 (PUT);

#### **3º parte**

- c. Registo da atribuição feita na tabela “campo\_semana” (POST).

## Guião do Cenário 2:

### 1º parte

1. No perfil (cliente), clicar no botão “Pedir para ser Monitor”;



Fig.20– Enviar Pedido para tornar-se Monitor (Cenário 2).

### 2º parte

2. No perfil (admin), clicar no botão “Ver Pedidos de Monitor”;

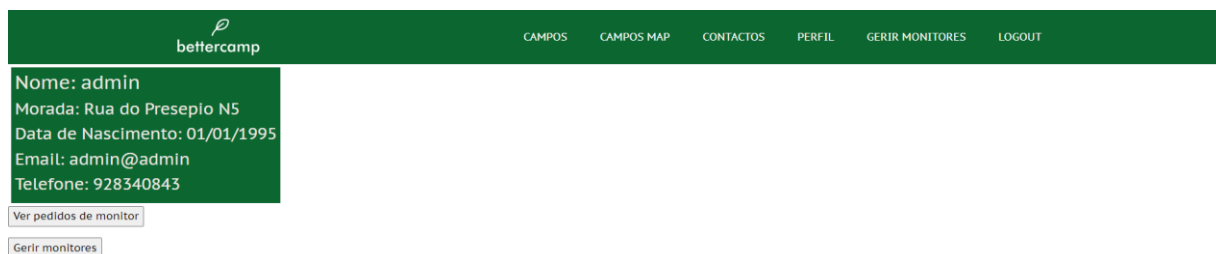


Fig.21– Aceder a página de pedidos (Cenário 2).

3. Apresentação dos pedidos feitos pelos clientes;



Fig.22– Pedidos feitos por clientes (Cenário 2).

4. Aceitar pedido, clicar no botão “Aceitar o Monitor”;



Fig.23– Pedido aceite “Elsa Fialho Pinto” (Cenário 2).

### 3º parte

#### 5. No perfil (*admin*), clicar no botão “Gerir Monitores”;

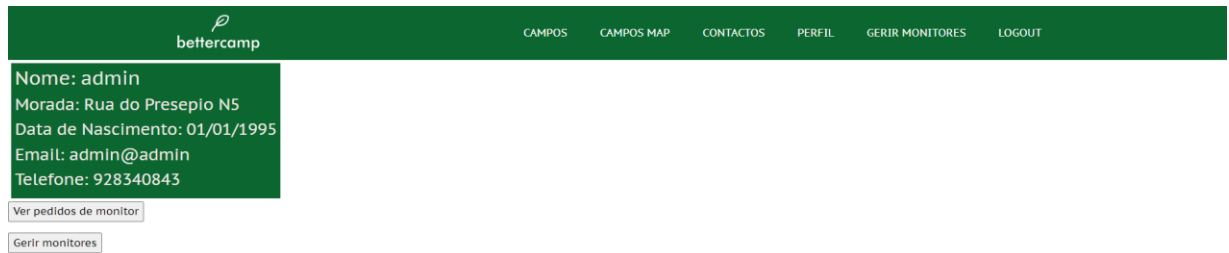


Fig.24– Aceder a página de gestão de monitores (Cenário 2).

#### 6. Submeter associação depois de dados inseridos corretamente;

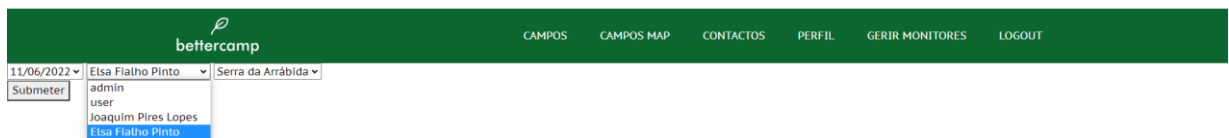


Fig.25– Submeter associação (Cenário 2).

### 3. Apresentação de locais de inscrição presencial (Contactos) / Rotas até Campos:

#### Cenário Principal:

##### 1º Parte

1. Na homepage aceder á tab de “Contactos”;
2. Permitir partilhar a localização no pop-up do navegador;
3. Na página vai ser apresentado todos as unidades/centros de saúde disponíveis ao lado esquerdo;
  - a) É feito um método GET para ser apresentado todos os centros de saúde.
4. Mapa apenas apresenta os campos num raio de 3.6km (3600m) da posição atual do utilizador.
  - a) É utilizado uma *query* de SQL para filtrar os campos pela distância e é feito um segundo método GET, com recurso à *query*.

##### 2º Parte

5. Na barra de navegação aceder a “Campos *Map*”;
6. Permitir a partilha de localização caso não feito anteriormente;
7. No mapa são apresentados os campos, pop-up do campo apresenta um botão de “Direções”;
  - a) É feito um método GET para ser apresentado todos os campos, para as rotas é utilizado uma função do *MapQuest* de direções e rotas.

## Guião do Cenário 3:

### 1º Parte

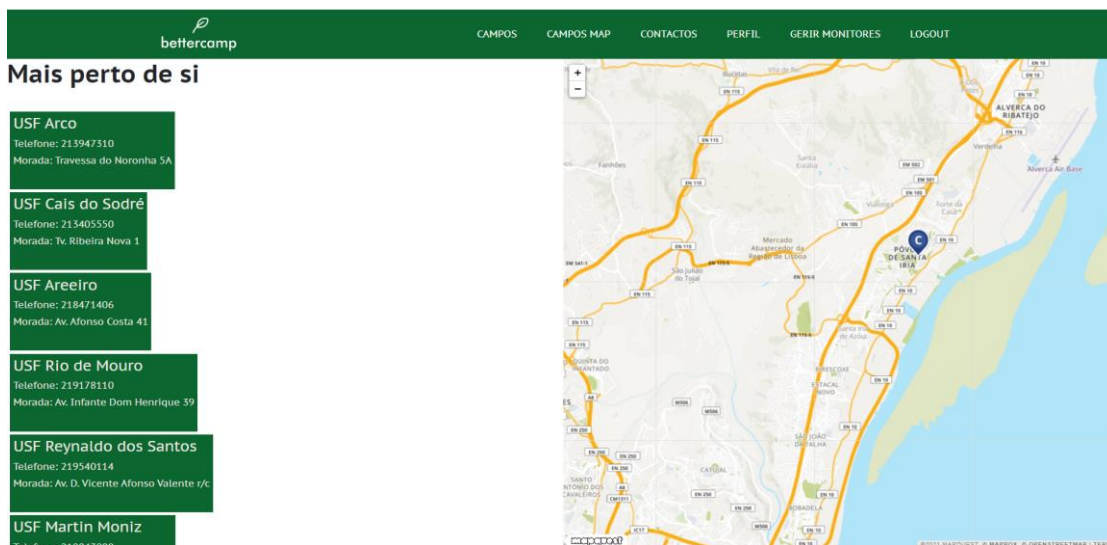


Fig.26– Centros de Saúde e mapa com mais próximos (Cenário 3).

### 2º Parte

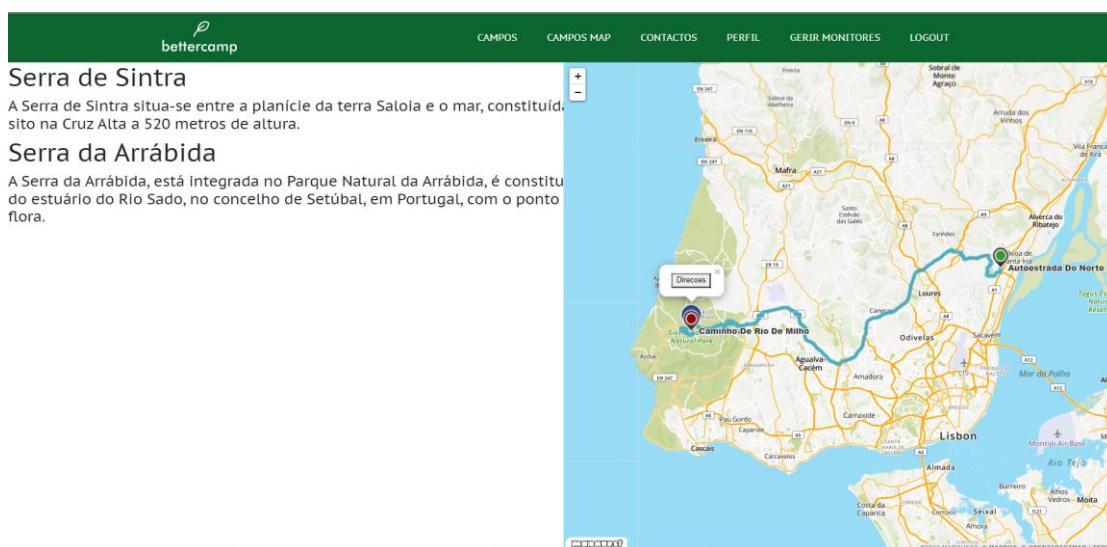


Fig.27– Rota de utilizador ao Campo (Cenário 3).



## Documentação REST

### Ir buscar todos os admins

Devolve todos os admins presentes na base de dados

Url: /api/admin (get)

### Ir buscar os dados de um admin específico

Devolve o os dados do admin com o Id escolhido

Url: /api/admin/:id (get)

Parâmetros URL

id – Integer que corresponde ao id do admin logado

Error Response

Code: 404

Content: {msg: "Pessoa not found!"}

### Aceitar um monitor dando update numa coluna

Dá update numa das colunas da tabela monitor\_insc

Url: /api/admin/aceitarMonitor (put)

Data Params

let obj = {"5"}

### Ir buscar todos os pedidos feitos pelos clientes para se tornarem monitor

Devolve todos os pedidos feitos pelos clientes

Url: /api/admin/:id/pedidosMonitor (get)

Parâmetros:

id – Integer que corresponde ao id do admin logado

### Ir buscar todos os campos

Devolve todos os campos presentes na base de dados

Url: /api/campos (get)

<b>Ir buscar um campo em específico</b>
Devolve o os dados do campo escolhido
Url: /api/campos/:id (get)
Parâmetros URL id – Integer que corresponde ao id do campo escolhido
Error Response Code: 404 Content: {msg: "Campo not found!"}

<b>Registar um monitor numa certa semana e campo</b>
Insere a semana, o campo e o monitor numa tabela da base de dados
Url: /api/admin/:id/gerirMonitor (post)
Parâmetros URL id – Integer que corresponde ao id do admin logado
Data Params let obj = { semana: "2", monitor: "9", campo: "1", };
Success Response Code: 200 Content: {msg: "Pedido efectuado com sucesso"}

<b>Fazer a reserva de um campo</b>
Insere o campo, a semana e a pessoa numa tabela da base de dados
Url: /api/campos/:id/reservas (post)
Parâmetros URL id – Integer que corresponde ao id do campo escolhido.
Data Params let obj = {"1", "12", "2"};
Success Response Code: 200 Content: {msg: "Inscrição efectuada com sucesso"}
Error Response Code: 401 UNAUTHORIZED Content: {msg: "Já está inscrito"}

<b>Ir buscar as atividades de cada campo</b>
Devolve as atividades de um campo em específico
Url: /api/campos/:id/ativ (get)
Parâmetros URL id – Integer que corresponde ao id do campo escolhido.

<b>Ir buscar todos os centros de saúde</b>
Devolve todos os centros de saúde
Url: /api/centros (get)

<b>Ir buscar os centros de saúde mais próximos</b>
Devolve os centros de saúde mais próximos da localização atual do utilizador
Url: /api/centros/proximos (get)

<b>Ir buscar todos os monitores</b>
Devolve todos os monitores presentes na base de dados
Url: /api/monitor (get)

<b>Ir buscar todas as semanas</b>
Devolve todas as semanas em que existem campos
Url: /api/semana (get)

<b>Ir buscar todas as pessoas</b>
Devolve todas as pessoas presentes na base de dados
Url: /api/pessoas (get)

<b>Ir buscar os dados de uma pessoa específica</b>
Devolve o os dados da pessoa com o Id escolhido
Url: /api/pessoa/:id (get)
Parâmetros URL:

id – Integer que corresponde ao id da pessoa logado
Error Response Code: 401 UNAUTHORIZED Content: {msg: "Wrong email or password"}

<b>Envia o email e a password da tentativa de login</b>
Verifica se os dados estão na base de dados
Url: /api/pessoa/login (post)
Data Params let obj = { email:" 20190908@iade.pt", pass: "12345", };
Error Response Code: 404 Content: {msg: "Pessoa not found!"}

<b>Submete o pedido para ser monitor</b>
Insere o id da pessoa numa tabela da base de dados
Url: /api/pessoa/pedirMonitor (post)
Data Params let obj = {"5"}
Success Response Code: 200 Content: {msg: "Pedido efectuado com sucesso"}

<b>Registar uma pessoa na plataforma</b>
Inserir todos os dados necessários na base de dados
Url: /api/pessoa/register (post)
Data Params <pre>let inputData = {   nome: "João",   morada: "Iade",   dtnasc: "2001/06/11",   genero: "M",   email: "20190908@iade.pt",   pass: "12345",   tlm: "912345678", };</pre>
Success Response Code: 200 Content: {msg: "registado com sucesso"}
Error Response Code: 401 UNAUTHORIZED Content: {msg: "Já está registado"}

# Sistemas Informação Geográficos/Programação Web

No nosso projeto decidimos implementar várias etapas como combinado durante as aulas, essas etapas incluíam existir dados geográficos na base de dados relativamente à posição dos campos e também na vertente da página de contactos com pontos de USF (Unidades de Saúde Familiares) e centros de saúde.

Na base de dados começamos por criar tabelas em que iriam levar um *populate* com esses pontos geográficos como podemos ver na figura 28 (*Create*) e na figura 29 (*Populate*).

```
create table centro_saude (centro_id int not null auto_increment,  
                           nome varchar (50),  
                           centro_tlm int,  
                           centro_morada VARCHAR(100),  
                           coords POINT not null SRID 4326,  
                           primary key (centro_id));
```

Fig.28– *Create table* (centros de saúde).

```
INSERT INTO centro_saude (nome,centro_tlm,centro_morada, coords) VALUES ('USF Cais do Sodré',213405550,'Tv. Ribeira Nova 1', ST_GeomFromText('POINT(38.70725188152147 -9.145017836093594)', 4326));  
INSERT INTO centro_saude (nome,centro_tlm,centro_morada, coords) VALUES ('USF Areeiro',218471406,'Av. Afonso Costa 41', ST_GeomFromText('POINT(38.739812839502854 -9.126703363148232)', 4326));
```

Fig.29– *Populate* (centros de saúde).

No primeiro mapa na página dos “Campos *Map*”, estamos a ir buscar os pontos geográficos diretamente na tabela campos, apresentamos utilizando o *MapQuest* um mapa com vários tiles, os *markers* são criados com pop-ups para mostrar um botão de direções (Figura 30).

```
if (navigator.geolocation) {
  navigator.geolocation.getCurrentPosition((position) => {
    latlong = [position.coords.latitude, position.coords.longitude];

    map = L.mapquest.map("map", {
      center: latlong,
      layers: L.mapquest.tileLayer("map"),
      zoom: 10,
    });

    // obterDirecoes(38.79770117227494, -9.432511769793297);

    for (let campo of campos) {
      console.log(campo.campo_lat, campo.campo_long)
      L.marker([campo.campo_lat, campo.campo_long], {
        icon: L.mapquest.icons.marker({
          primaryColor: "#22407F",
          secondaryColor: "#3B5998",
          shadow: true,
          size: "md",
          symbol: "C",
        }),
      })
        .bindPopup(popUp)
        .addTo(map);
    }
  })
}
```

Fig.30– Criação do *mapa* e *markers* (CamposMap.js).

Usamos a API de direções e rotas diretamente sobre o mapa, passamos a localização já antes guardada usando um *sessionStorage*, e a localização do destino. (Figura 31).

```
function popUp(layer) {
  let btn = document.createElement("button");
  btn.innerHTML = "Direcoes";
  btn.onclick = function () {
    L.mapquest
      .directions()
      .route({
        start: latlong,
        end: layer._latlng,
      })
  }

  return btn
}
```

Fig.31– Rotas e direções (CamposMap.js).

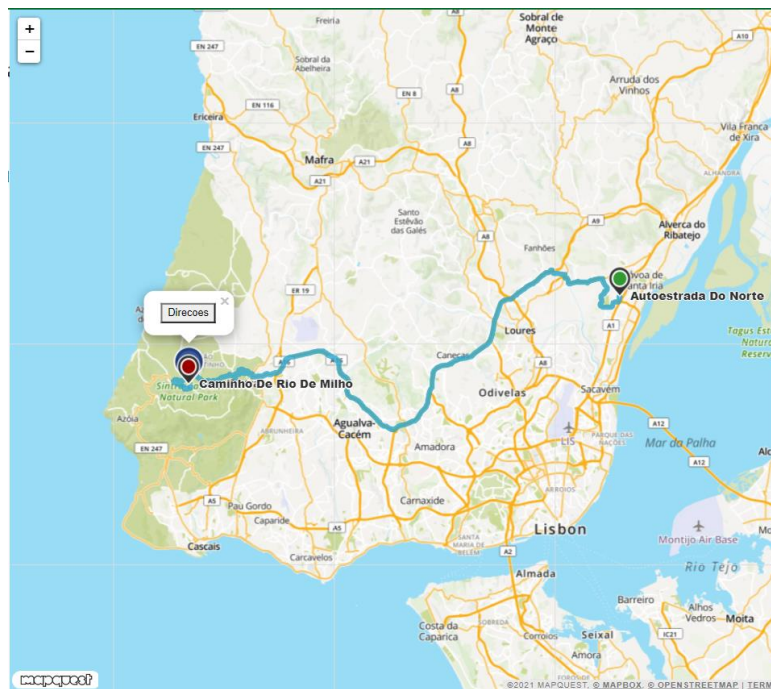


Fig.31– Mapa Rotas (Campos Map).

No segundo mapa na página dos “Contactos”, decidimos mostrar todos os centros de saúde ou USF (Unidades de Saúde Familiar) que temos na base de dados como já mostrado antes. Criar os markers e meter os mesmos no mapa não seria nada de novo como já tínhamos o primeiro mapa, então decidimos utilizar um filtro predefinido na busca dos campos.

Este filtro é feito usando os pontos geográficos na tabela dos centros, mas compara com a localização do utilizador que abre a página, e procura que pontos estão numa área de 60x60. É feito usando recurso a uma query de SQL como podemos ver na figura 32.

```
module.exports.getAllCentrosProximos = async function (lat, long) {
  try {
    let sql = `SELECT * FROM centro_saude WHERE ST_Distance_Sphere(coords, ST_GeomFromText("POINT( ? )", 4326 )) <=(60*60);`;
    let result = await pool.query(sql, [parseFloat(lat), parseFloat(long)]);
    let centro = result;
    return { status: 200, result: centro };
  } catch (err) {
    console.log(err);
    return { status: 500, result: err };
  }
};
```

Fig.32– Query SQL (Functions.js).



```

var router = express.Router();
var mProd = require("../models/functions.js");

router.get("/", async function (req, res, next) {
  let result = await mProd.getAllCentros();
  res.status(result.status).send(result.result);
});

router.get("/proximos", async function (req, res, next) {
  let lat = req.query.lat;
  let long = req.query.long;
  console.log(lat, long)
  let result = await mProd.getAllCentrosProximos(lat, long);
  res.status(result.status).send(result.result);
});

module.exports = router;

```

Fig.32– Route (centros.js).

Quando carregada a página dos contactos e depois de ser fornecida a localização do utilizador, conseguimos passar os parâmetros pelo URL da API como conseguimos ver pela figura 33.

```

let obj = {lat,long};
console.log(obj)
let centros = await $.ajax( {
url: ` /api/centros/proximos?lat=${lat}&long=${long}`,
method: "get",
dataType: "json",
});

```

Fig.33– Contactos.js

**OBS:** Para testes poderá ser mudado os valores de área na *query* caso não mostre nada aos professores, visto que temos alguns centros, mas não os suficientes para num raio de 3.6 km aparecer.