

**Universidade do Minho**

Escola de Engenharia

Mestrado Integrado em Engenharia Informática

**Unidade Curricular de**

**Laboratórios de Informática IV**

Ano Letivo de 2017/2018

**GuguDadah**

**Carlos Campos A74745**

**Diana Costa A78985**

**Marcos Pereira A79116**

**Sérgio Oliveira A77730**

**Vítor Castro A77870**

Março de 2018

**LI41617**

|  |  |
| --- | --- |
| Data de Recepção |  |
| Responsável |  |
| Avaliação |  |
| Observações |  |

**GuguDadah**

**Carlos Campos A74745**

**Diana Costa A78985**

**Marcos Pereira A79116**

**Sérgio Oliveira A77730**

**Vítor Castro A77870**

Março de 2018

# Resumo

Este relatório descreve a primeira etapa do projeto “GuguDadah”, que surge da necessidade de implementar uma solução informática que permita gerir um serviço de babysitting ao domicílio, e respetivos babyssitings.

A primeira fase consiste na fundamentação do projeto, onde o mesmo será contextualizado, feita a definição da identidade do sistema e explicitadas as motivações e respetivos objetivos. Nestes tópicos, será descrito o enquadramento do projeto e seu modelo de funcionamento, tal como os principais razões que levam à implementação e propósitos a alcançar no término do mesmo.

De seguida, é feita uma análise de viabilidade, dos recursos necessários e maquete do sistema. Assim, é possível estabelecer o custo aproximado do projeto, bem como descrever a forma como vão ser obtidos os dados para a aplicação e representar a arquitetura do sistema idealizado. Identificam-se também os principais constituintes do sistema e respetivas interações.

Fechando o primeiro capitulo, é obtido um conjunto de medidas de sucesso a cumprir, bem como o plano de desenvolvimento. Com estes, será possível avaliar o programa final e a eficácia da equipa nas diversas fases do desenvolvimento.

**Área de Aplicação:** Desenvolvimento de um sistema de babysitting ao domicílio

**Palavras-chave:** Desenvolvimento de Software, Engenharia de Software, aplicação, babyssiting, infantário, localização, gestão de projeto, diagrama de Gantt.

# Índice

[1.1. Contextualização 1](#_Toc508041918)

[1.2. Motivação e Objetivos 2](#_Toc508041919)

[1.3. Definição da Identidade do Sistema a Desenvolver 3](#_Toc508041920)

[1.4. Análise de Viabilidade 4](#_Toc508041921)

[1.5. Identificação dos Recursos Necessários 6](#_Toc508041922)

[1.6. Modelo de Sistema 7](#_Toc508041923)

[1.7. Definição de Medidas de Sucesso 8](#_Toc508041924)

[1.8. Plano de Desenvolvimento 9](#_Toc508041925)

# Índice de Figuras

[Figura 1 – Maquete do Sistema 7](#_Toc508041913)

[Figura 2 - Diagrama de Gantt: Fundamentação 9](#_Toc508041914)

[Figura 3 - Diagrama de Gantt: Especificação 9](#_Toc508041915)

[Figura 4 - Diagrama de Gantt: Implementação 9](#_Toc508041916)

1. Definição do Sistema
   1. Contextualização

Em pleno século XXI, com o desenvolvimento industrial e tecnológico, cresce a necessidade de produção contínua e rápida por parte das empresas, para singrar nos mercados económicos. Com isso aumenta a carga horária dos trabalhadores, desde o nível mais laboral, até ao executivo. Tudo isto gera um problema sério para quem tem filhos, visto que cria um défice de tempo livre e, por vezes, com reuniões até tarde ou a condição de trabalhar horas extra, torna-se difícil arranjar alguém para tomar conta dos filhos. Por vários motivos, ora os avós moram longe, ora o casal amigo tem um jantar importante, e como os infantários não trabalham até horas tardias, é causado o pânico.

A empresa Bebés & Companhia, que possui uma cadeia de infantários espalhados pela grande Lisboa, tendo em vista essa adversidade, com o reparo que a maioria dos seus funcionários estão na faixa etária dos vinte, viram aqui uma ótima oportunidade. Surge, assim, a ideia de fazer um serviço ao domicilio de babysitting.

Depois da realização de um inquérito por todos os jardins de infância, em que foram abordadas várias questões, como a disposição para trabalhar horas extra, disponibilidade para horas tardias e para deslocações ao domicílio, foi concluído que a adesão foi de acordo com as expectativas a nível de idades e com isso ficaria acessível um grande volume de mão-de-obra, por parte da própria empresa, não havendo a intenção de contratar agentes externos.

Quando a equipa de desenvolvimento de software foi contactada, após algumas reuniões e um estudo do mercado, ficou acordado que a ideia seria patenteada e que a mesma seria viável e suscetível a grande escalabilidade. Seria assim criada a plataforma “GuguDadah”.

* 1. Motivação e Objetivos

O principal motivo que levou o grupo à realização deste projeto, é fundamentalmente a dificuldade de encontrar alguém que seja responsável para tomar conta dos nossos filhos a horas tardias, ou a falta de disponibilidade dos infantários para cuidar de crianças adoentadas. Seja por causa de uma reunião, ou também por causa de um jantar muito importante, por vezes não existe auxílio por parte de algum ente querido, e como não existe por norma infantários abertos em horário noturno, o problema é agravado. A política atual dos infantários também inclui que crianças doentes, para não afetar as restantes, deverão ficar em casa.

Outro fundamento para esta motivação, é a inexistência de um sistema que permita a requisição de serviços de babysitting online, em Portugal. O mais próximo que poderemos encontrar são os anúncios no OLX, e relembrando que este é essencialmente um site de venda de produtos em segunda mão. Assim sendo, não existe um controlo da atividade prestada, nem qualquer tipo de garantia ou de qualidade por parte da pessoa que presta a função exigida.

Uma condição que suporta a iniciativa da equipa é que a maioria dos trabalhadores dos Infantários Bebés & Companhia possuem idade no intervalo dos vinte aos trinta. Como faz parte do senso comum, cidadãos nestas idades têm tendência para querer angariar mais fundos monetários, visto que estão em inícios de vidas autónomas, e como a maioria também não possuiu crianças, faz com que tenham uma maior disponibilidade e vontade, para se prontificaram a trabalhar a horas tardias, e assim são um grande suporte a esta ideia de negócio.

Com uma ideia original, que é apoiada por estes vários fatores, foi tomada a decisão de fazer um estudo do mercado para consolidar a ambição do grupo, que contactou a “MarkUp”, uma empresa especializada em fazer esse tipo de análise. Depois de realizada a devida observação do relatório final, concluiu-se que este projeto iria obter sucesso, seguindo várias condicionantes que foram referidas anteriormente.

* 1. Definição da Identidade do Sistema a Desenvolver

O projeto a desenvolver pela equipa, denominado "GuguDadah", é um sistema de gestão de serviços de babysitting personalizados, oferecido pelo infantário "Bebés & Companhia", que decidiu expandir o seu serviço diurno e localizado para um que funcionasse non-stop e ao domicílio. Os pais das crianças, quer inscritas no infantário, quer externas, poderão requisitar um educador profissional e certificado para se deslocar a sua casa, em qualquer horário, conforme o tipo de préstimo que optaram. Os profissionais serão avaliados pelos pais conforme os seus serviços anteriores, e os pais obterão descontos de acordo com a frequência com que requerem préstimos ou caso tenham filho(s) inscritos no "Bebés & Companhia". Por último, é de salientar que um educador tem sempre acesso à navegação por GPS sempre que estiver a executar a rota até ao domicílio combinado. Resumem-se de seguida algumas características identificativas do sistema a desenvolver, com um objetivo meramente elucidativo, uma vez que as mesmas serão desenvolvidas à frente no relatório.

|  |  |
| --- | --- |
| Nome: | GuguDadah |
| Slogan: | De Pais, Para pais |
| Categoria: | Babysitting |
| Características: | Implementação de descontos em serviços |
| Reserva/Requerimento de profissionais para serviço |
| Avaliação dos profissionais |
| Navegação GPS |
|  |
| Faixa Etária: | 20-50 |
| Logótipo: | logo |

* 1. Análise de Viabilidade

De maneira a analisar a viabilidade deste novo serviço, coube ao grupo cumprir o papel de adicionar, sobre a análise de mercado inicial da empresa contratante, uma outra análise ou estudo da adequação deste empreendimento às circunstâncias do serviço ao domicílio.

Um sistema de entregas ao domicílio 24/365 é um modelo de negócio cada vez mais comum no tempo presente, que demonstra uma abertura do cliente à adaptação contínua e adoção de novas formas de comércio. A popularidade deste modelo indica, à partida, a viabilidade da solução. Foi realizada, assim mesmo, uma pesquisa breve relativa à presença destes tipos de negócio no mercado atual.

Constatou-se rapidamente que a adoção das entregas ao domicílio é cada vez mais geral, havendo anúncios da abertura destes serviços por parte de grandes marcas com uma frequência elevada, como a *Amazon*, que melhora continuamente as suas opções de entrega. Seguem-se alguns outros exemplos:

* "*The owner of KFC and Taco Bell is teaming up with Grubhub to expand home delivery [...]" (8 Fev*, 2018)
* "*McDonald's cheers home delivery success as UK spurs sales rise*" (Jan 30, 2018)
* "*UberEats Starts Delivering Food in Detroit Today*" (Nov 30, 2017)
* "*Burger King já faz entregas ao domicílio em Lisboa*" (Jan 6, 2017)

De uma forma mais geral, o termo de pesquisa "home delivery" tem visto um crescimento estável já há vários anos, sendo por isso um alvo seguro para um negócio que já esteja estabelecido e queira aumentar as suas chances de se manter relevante numa sociedade que está continuamente em desenvolvimento.

Foi demonstrado acima, de maneira resumida, que o modelo de negócio projetado é popular. No entanto, um leitor atento pode já ter reparado que o tipo negócio que mais participa nesta adoção tende a estar relacionado com a restauração.

A criação de um serviço de babysitting ao domicílio posiciona-se, então, como uma opção de entrada num mercado estável e de crescimento contínuo, mas ainda assim inovadora o suficiente para se manter um movimento original, mais do que um investimento trivial.

Foram feitos estudos de mercado com recurso a inquéritos por telefone, que demonstraram uma elevada recetividade à possibilidade de agendar um(a) baby-sitter a qualquer altura do dia, durante todo o ano. O facto de o serviço estar associado a um lugar físico onde se podem conhecer os profissionais, aumenta também a confiança por parte dos pais.

Foi também descoberto que a perda de confiança comum ao movimento inicial de um serviço para um meio digital é compensada pela presença de avaliações pessoais e rankings no sistema, que aumentam o nível de transparência face ao consumidor.

Uma posterior avaliação da proposta por parte de um comité de gestores concluiu que o orçamento estipulado é adequado ao nível de complexidade do projeto. Havendo a possibilidade de manutenção a longo termo, e tendo em conta as projeções positivas da receção do novo modelo de negócio, este empreendimento é dado como uma mais valia para a organização à qual o grupo pertence.

* 1. Identificação dos Recursos Necessários

Para este projeto estima-se a necessidade de alocação de um engenheiro de software e quatro programadores. O engenheiro de software irá assumir também o papel extraordinário de gestor de projeto e de gestor de bases de dados, para além de ser responsável pela especificação do projeto. Os programadores irão trabalhar 2h/dia, daí a necessidade de quatro profissionais envolvidos no projeto, que dá o equivalente a 1homem/mês.

Não serão utilizados quaisquer equipamentos para a realização deste projeto, para além dos computadores disponibilizados pela software house associada aos engenheiros e programadores.

Numa fase inicial serão feitas duas reuniões com CEO do infantário Bebés & Companhia. Na primeira vai-se proceder ao levantamento de requisitos e, no fim do período de planeamento do projeto, a confirmação do modelo elaborado. Ambas as reuniões desta fase serão feitas nas instalações da empresa que pretende contratar o serviço, exigindo um veículo ligeiro para deslocação da equipa em ambas as ocasiões. No intervalo das reuniões será também necessário fazer um estudo de mercado, encomendado a uma empresa externa.

A equipa será alocada ao projeto durante três meses, com algumas interrupções para participação em outros projetos. Por este motivo, é natural que surjam períodos de pausa no desenvolvimento.

É também necessário alocar um servidor que estará na software house e que será o responsável pelo alojamento dos serviços 24/7 assim como se prevê a inevitabilidade de se adquirir tablets para a equipa realizar testes nas diferentes plataformas existentes no mercado.

Os diferentes softwares e suas licenças estarão, à partida, garantidos pelos trabalhadores da empresa pelo que não acarretam nenhum encargo financeiro para a empresa.

Por fim, o projeto será desenvolvido inteiramente nos escritórios da empresa, levando à ocupação dos espaços habituais dos colaboradores.

* 1. Modelo de Sistema

O sistema consiste num servidor web ASP.NET ligado a uma base de dados SQL Server, onde serão guardados todos os dados dos intervinientes no sistema. O servidor expõe uma API pública que é disponibilizada através do site, assim como uma API privada que é utilizada pela aplicação móvel.

O site web é usado por clientes de maneira a escolher baby-sitters, agendar trabalhos, e submeter avaliações. A aplicação móvel é usada pelos trabalhadores e permite a verificação do perfil profissional e dos trabalhos agendados, indicando também o caminho até estes (deslocação GPS até ao domicílio pretendido).

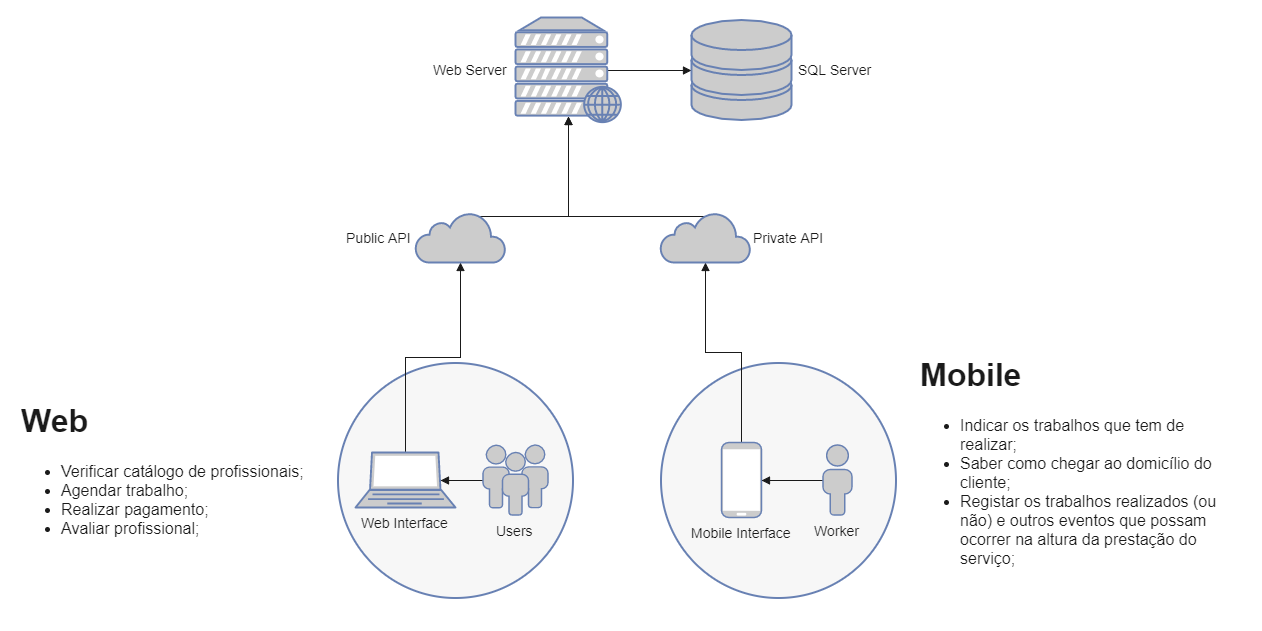


Figura 1 – Maquete do Sistema

* 1. Definição de Medidas de Sucesso

Para proceder a uma avaliação de qualidade do projeto definiram-se, desde logo, um conjunto de pontos a verificar. Assim, segue a lista com os tópicos a cumprir para uma boa execução do plano traçado:

* A aplicação deve ser classificada pelos utilizadores como de fácil utilização;
* Devem ser atingidos 1000 usos da aplicação no período de 365 dias;
* O nível de satisfação dos utilizadores do infantário, avaliado através dos inquéritos anuais, deve subir, pelo menos, 3 pontos percentuais;
* O número de utilizadores do infantário deve subir 5 pontos percentuais, um ano após a implementação;
* A aplicação deve funcionar 24h por dia, 365 dias por ano.
  1. Plano de Desenvolvimento

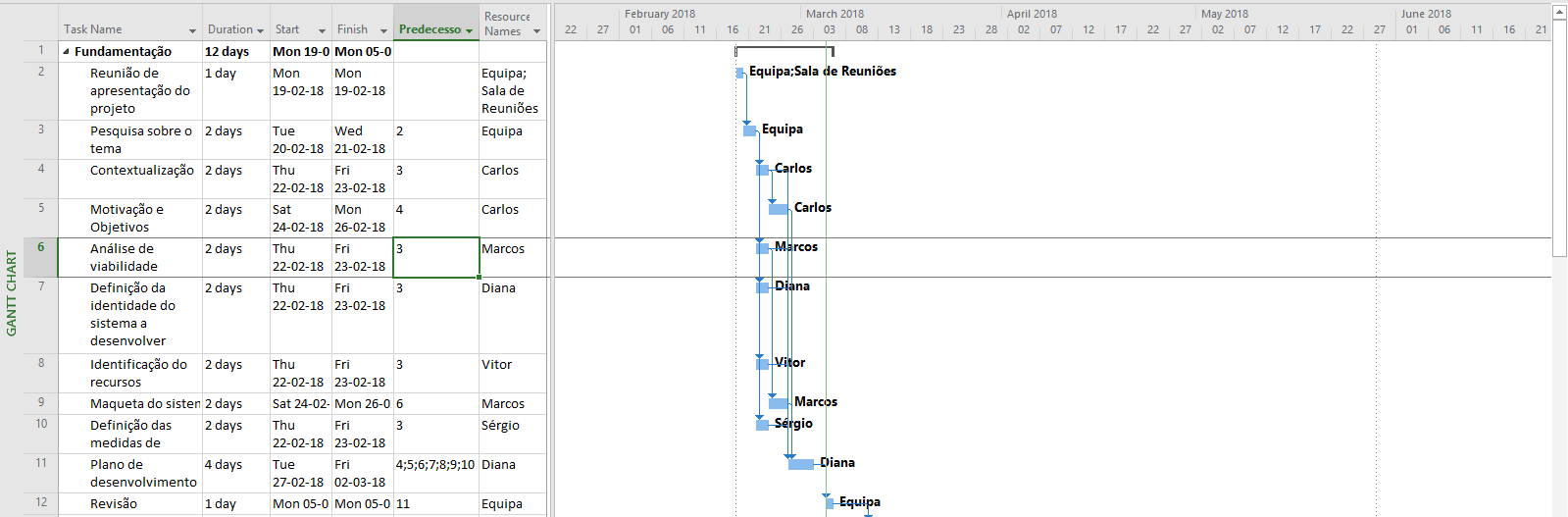


Figura 2 - Diagrama de Gantt: Fundamentação

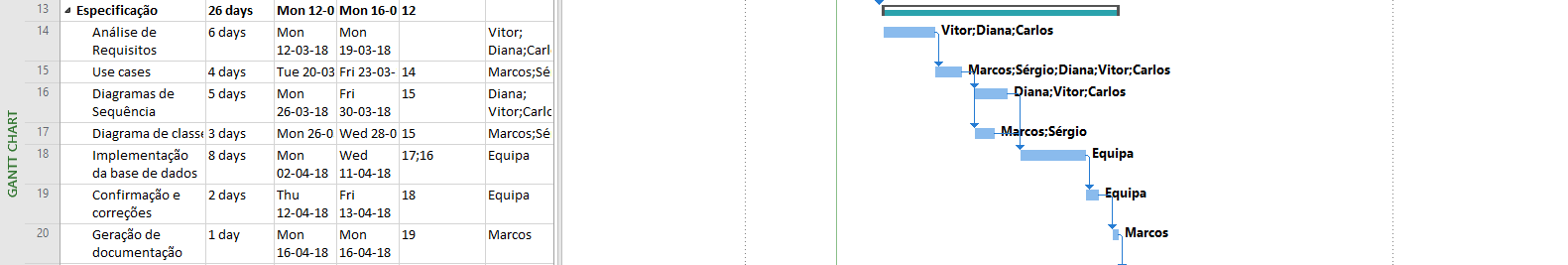


Figura 3 - Diagrama de Gantt: Especificação

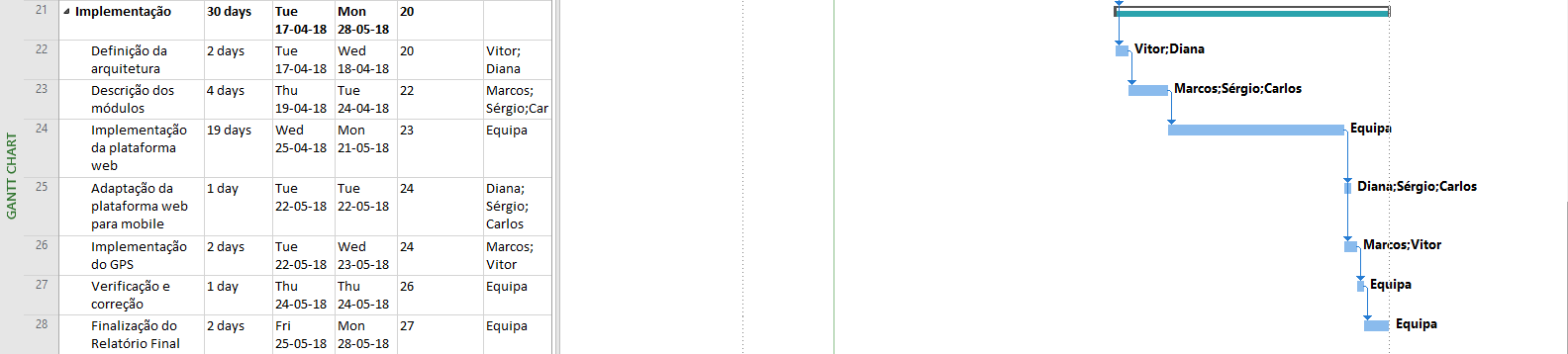


Figura 4 - Diagrama de Gantt: Implementação

Na primeira fase foi feita a fundamentação do tema. Após feita uma intensiva pesquisa inicial, com vista à contextualização correta com o problema, foi possível fazer a divisão das diversas tarefas de acordo com os pontos fortes dos elementos. Assim, apenas a fase de pesquisa sobre o tema e de revisão ficaram como trabalho de equipa.

O tempo dado a capa um dos contribuidores, apesar de por vezes poder ser maior (não tendo implicações no prazo de entrega da primeira parte), vai de acordo com as estimativas do gestor de projeto.

Nas fases seguintes haverá a necessidade de ter toda a equipa reunida em torno de uma tarefa ou outra tarefa, dada a importância da mesma. Todas as outras serão distribuídas segundo o mesmo critério que na primeira fase.

2 FASE

1. Modelação em UML
   1. Diagramas de sequência

Por forma a melhor descrever o comportamento do sistema, principalmente em relação ao utilizador, desenvolveram-se alguns diagramas de sequência. Estes, que aprofundam os Use Cases especificados, descrevem as funcionalidades do sistema a encontrar a cada passo.

Este tipo de diagrama, pela sua tipologia de interação Utilizador-Sistema, permite a aproximação e perceção das necessidades relativas à fase de desenvolvimento. É assim tornado claro para a equipa de desenvolvimento os componentes e funcionalidades a desenvolver.

* + 1. Exemplo de Diagrama de Sequência - Aceitar proposta de trabalho

De forma a explicitar o Use Case fornecido anteriormente, dado pelo nome de “Aceitar proposta de trabalho”, é apresentado, de seguida, o seu diagrama de sequência.

Neste diagrama é possível verificar que o ator corresponde ao profissional, que usará a aplicação. Tal como explicitado, para um funcionário aceitar determinada proposta de trabalho, é exigida ao sistema uma obtenção de todas as propostas de trabalho à espera de aceitação, para aquele funcionário. Após esta obtenção, e caso haja propostas, deverá ser selecionada a proposta a aceitar, sendo que sistema verificará se esta ainda está disponível. Não sendo possível aceitar este trabalho (por eventual cancelamento), o funcionário é informado desta situação e termina a interação. O funcionário pode também escolher não aceitar a proposta de trabalho, sendo que esta informação surgirá informada ao cliente. Caso tudo corra bem, o sistema guardará a aceitação do trabalho e informará o funcionário que já o pode consultar junto dos seus trabalhos agendados.

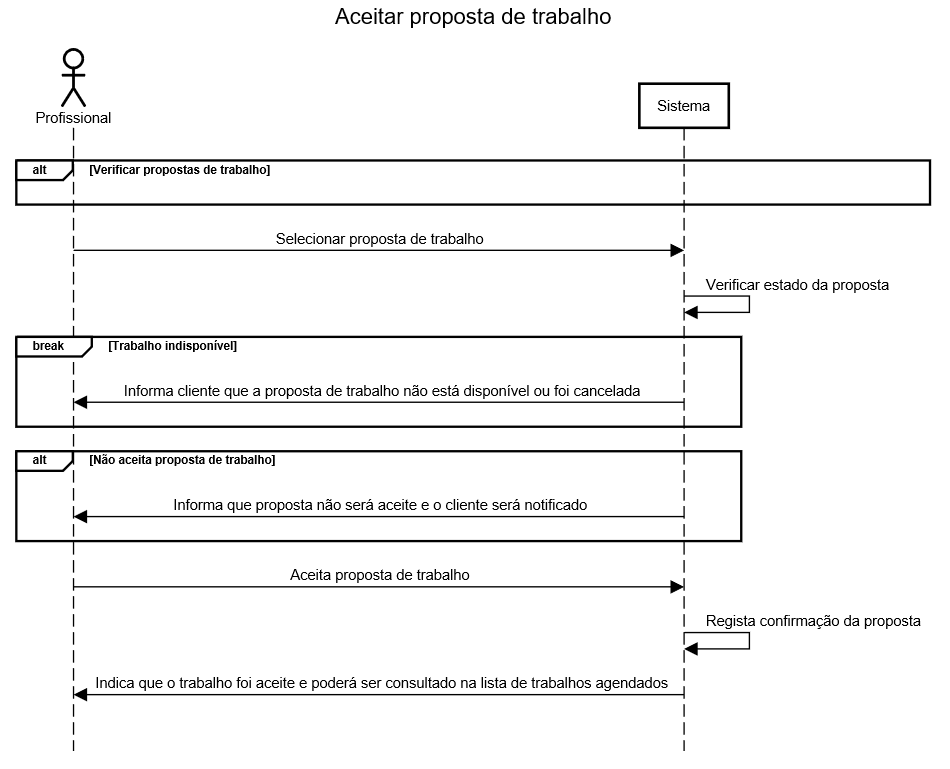


Figura 5 - Diagrama de Sequência "Aceitar Proposta de Trabalho"

* + 1. Exemplo de Diagrama de Sequência - Avaliar trabalho

Em sequência com o Use Case fornecido anteriormente, dado pelo nome de “Avaliar trabalho”, é apresentado, de seguida, o seu diagrama de sequência.

Neste diagrama é possível verificar que o ator corresponde ao cliente, que usará a aplicação. Tal como explicitado, para um cliente avaliar determinado trabalho, é exigida ao sistema uma obtenção de todos os trabalhos já efetuados, por ordem daquele cliente. Após esta obtenção, e caso haja trabalhos já efetuados, deverá ser selecionado aquele a avaliar, sendo que sistema verificará se este já terá ou não sido avaliado. Não sendo possível proceder à avaliação (por já ter sido feita), o cliente é informado desta situação e termina a interação. Caso tudo corra bem, o ator poderá inserir a avaliação (0-5), sendo que o sistema regista-la-á, informando o sucesso do procedimento.

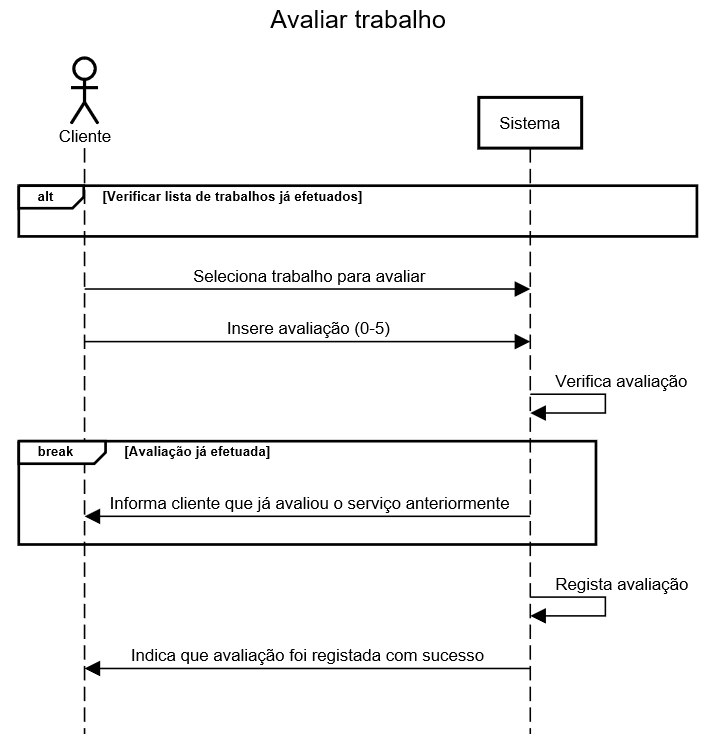


Figura 6 - Diagrama de Sequência "Avaliar Trabalho"

* + 1. Exemplo de Diagrama de Sequência - Registo

Por fim, e de acordo com o Use Case fornecido anteriormente, dado pelo nome de “Registo”, é apresentado, de seguida, o seu diagrama de sequência.

Neste diagrama é possível verificar que o ator corresponde ao cliente ou funcionário, que usará a aplicação. Tal como explicitado, para um ator fazer o seu registo, deve inserir o email, username e respetiva password. Após confirmados os dados inseridos, podem surgir 3 situações. Na primeira, os dados inseridos são insuficientes, sendo que a interação termina. Na seguinte, pode ser verificado que o utilizador já existe, sendo o ator notificado disto e abortado o processo de registo. Por último, e caso tudo corra bem, o utilizador será registado pelo sistema, que informará o sucesso do registo.

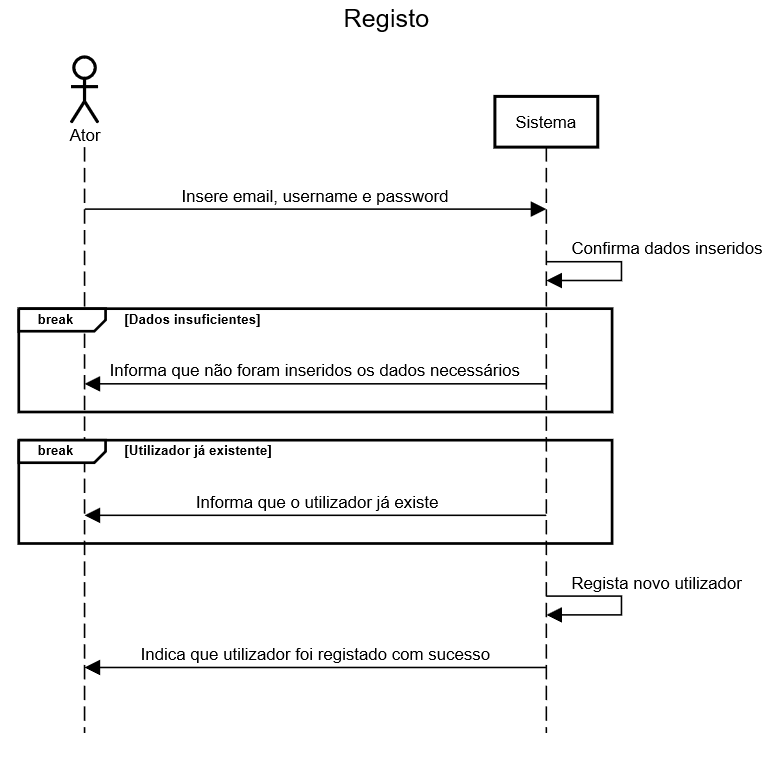


Figura 7 - Diagrama de Sequência "Registo"

* + 1. Exemplo de Diagrama de Sequência – Agendar Trabalho

Por fim, e de acordo com o Use Case fornecido anteriormente, dado pelo nome de “Agendar Trabalho”, é apresentado, de seguida, o seu diagrama de sequência.

Neste diagrama é possível verificar que o ator corresponde ao cliente, que usará a aplicação. Tal como explicitado, para agendar um serviço o cliente deverá selecionar data para o agendamento, bem como inserir a rua e localidade. Selecionará também a hora de início e fim, bem como possíveis serviços complementares, caso queira fazer uma atividade no exterior ou de estudo. Após confirmados os dados inseridos, podem surgir 4 situações de erro, que resultam no término da interação.

Neste momento, caso tudo tenha corrido bem, o utilizador poderá escolher um profissional, sendo essa escolha verificada pelo programa. Novamente, caso não tenha sido escolhido o profissional, a interação é abortada. Se tudo até ao momento foi validado, será apresentado o custo do serviço ao cliente, que o poderá aceitar ou não, efetuando o pagamento e confirmando o agendamento ou, caso não aceite aquele valor, cancelar o agendamento.

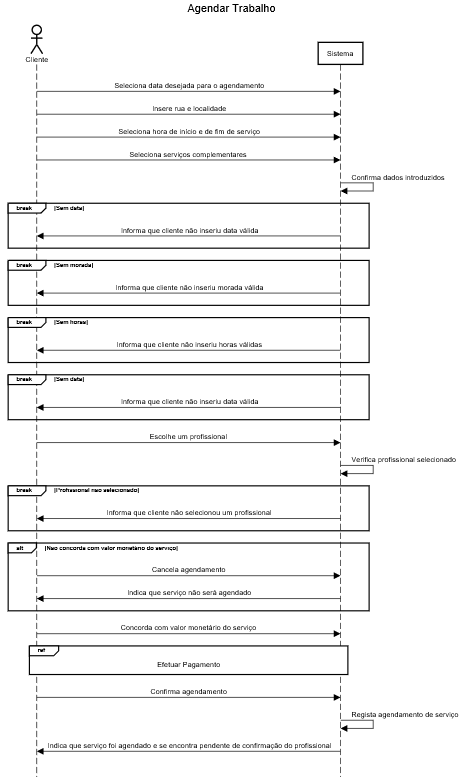


Figura 8 - Diagrama de Sequência "Agendar Trabalho"

1. Base de Dados
   1. Modelo Conceptual
      1. Identificar os tipos de identidades

Após a leitura e interpretação dos requisitos, identificaram-se vários objetos chave. De entre estes, surgiram as entidades: cliente, trabalho e funcionário.

|  |  |  |  |
| --- | --- | --- | --- |
| Nome da Entidade | Descrição | Sinónimos | Ocorrências |
| Cliente | Cliente que pretende usar a aplicação e encontrar alguém que cuide do seu filho | Utilizador, Responsável | Um cliente pode selecionar um cuidador para, em determinadas horas, cuidar do filho |
| Trabalho | Trabalho prestado a um cliente por um dado funcionário, que regista toda a informação útil sobre o mesmo | Serviço | Um cliente pede um trabalho, a determinado funcionário |
| Funcionário | Funcionário que presta serviços de cuidado, a pedido do cliente, mediante sua disponibilidade | Cuidador, Babysitter | Um funcionário pode aceitar ou não um serviço que lhe é proposto |

* + 1. Identificar tipos de relacionamento

Uma vez definidas as entidades do sistema, torna-se imperativo estabelecer as relações entre as mesmas. Novamente, por leitura e interpretação dos requisitos, identificaram-se os relacionamentos e respetiva multiplicidade, apresentados na tabela seguinte:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome da Entidade | Multiplicidade | Relacionamento | Multiplicidade | Nome da Entidade |
| Cliente | 1 | Requisita | N | Trabalho |
| Trabalho | N | Tem | 1 | Funcionário |

* + 1. Identificar e associar atributos com os tipos de entidades e relacionamentos

Tendo cada umas das entidades um conjunto de informações importantes a reter, as quais foram especificadas pelo cliente, surge a seguinte tabela. Esta evidencia todos os atributos retidos em cada uma das entidades.

* + 1. Atributos simples/compostos

Após a leitura dos requisitos, chegou-se à conclusão que não existe a necessidade de ter quaisquer atributos compostos.

* + 1. Atributos derivados

Após o levantamento de requisitos e identificação dos atributos, foi possível verificar que o atributo Custo é derivado, para a entidade Trabalho. O seu cálculo é feito através de uma fórmula que conjuga Duração e Tipo (da entidade Trabalho), com Turno (da entidade Funcionário) e Estatuto (da entidade Cliente).

* + 1. Atributos multivalor

Da mesma análise referida nos pontos anteriores, é possível concluir que não existem atributos multivalor. É de notar que, caso fosse requisito haver vários contactos para o Cliente, surgiria um atributo multivalor Contacto.

* + 1. Associação entre atributos e entidades

Apresenta-se, de seguida, a relação de atributos e respetivas entidades.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Nome da entidade | Atributo | Descrição | Tipo | Nulo | Multivalor | Derivado | Composto |
| Cliente | Username | É o username, que o identifica | Caracteres | Não | Não | Não | Não |
| Nome | Nome do cliente | Caracteres | Não | Não | Não | Não |
| Email | Email com que o cliente se registou | Caracteres | Não | Não | Não | Não |
| Password | Password atual da conta do cliente | Caracteres | Não | Não | Não | Não |
| Estatuto | Condição que pode permitir acesso a descontos | Valor inteiro positivo | Não | Não | Não | Não |
| Contacto | Telemóvel que o cliente registou | Valor inteiro positivo | Não | Não | Não | Não |
| Avatar | Imagem representativa do cliente | FILESTREAM | Sim | Não | Não | Não |
| Trabalho | Id | Identificador daquele trabalho | Valor inteiro positivo | Não | Não | Não | Não |
| Tipo | Tipo de trabalho que vai ser desempenhado | Char | Não | Não | Não | Não |
| Duração | Duração do trabalho em causa | Valor inteiro positivo | Não | Não | Não | Não |
| Custo | Custo final do trabalho realizado | Valor inteiro positivo | Não | Não | Sim | Não |
| Data | Data e Hora do início do trabalho | DATETIME | Não | Não | Não | Não |
| Avaliação | Avaliação do trabalho, caso seja avaliado | Valor decimal positivo | Sim | Não | Não | Não |
| Pagamento | Tipo de pagamento efetuado (CC ou dinheiro) | Char | Não | Não | Não | Não |
| Estado | Estado atual do pedido de serviço | Char | Não | Não | Não | Não |
| Observações | Observações efetuadas ao serviço | Caracteres | Sim | Não | Não | Não |
| Morada | Morada de prestação de serviço | Caracteres | Não | Não | Não | Não |
| Funcionário | Username | É o username, que o identifica | Caracteres | Não | Não | Não | Não |
|  | Nome | Nome do funcionário | Caracteres | Não | Não | Não | Não |
|  | Email | Email com que o funcionário está registado | Caracteres | Não | Não | Não | Não |
|  | Password | Password atual da conta do funcionário | Caracteres | Não | Não | Não | Não |
|  | Turno | Turno em que funcionário se encontra disponível | Char | Não | Não | Não | Não |
|  | Rating | Rating atribuido ao funcionário | Valor decimal positivo | Sim | Não | Não | Não |
|  | Contacto | Telemóvel que o funcionário está registado | Valor inteiro positivo | Não | Não | Não | Não |
|  | Avatar | Imagem representativa do funcionário | FILESTREAM | Sim | Não | Não | Não |

* + 1. Determinar domínio dos atributos

Neste passo serão descritos os domínios dos atributos das várias entidades. Tornar-se-á evidente o tipo e os valores que os atributos tomarão, tal como o porquê de tal decisão.

* Cliente
  + Username: String, que poderá conter todo o tipo de caracteres. Com tamanho limitado a 10, por questões de eficiência.
  + Nome: String, que conterá o nome do cliente.
  + Email: String, que conterá o email de registo do cliente.
  + Password: String, que conterá a password do cliente. Como a aplicação não necessita de elevada segurança, esta ficará guardada na mesma Base de Dados que todas as outras informações.
  + Estatuto: Char, que identifica o cliente como sendo Gold ou não, dependendo se é utilizador do infantário.
  + Contacto: Int, que guarda o número do cliente.
  + Avatar: FILESTREAM, que guardará a imagem do cliente. É guardado diretamente na BD pois a aplicação é de pequena dimensão e diminui a carga de implementação. De facto, não há necessidade de ter um host de imagens e de fazer backups ao mesmo, ao invés de fazer apenas a toda a BD.
* Trabalho
  + Id: Int, identifica o trabalho requerido. Autoincremental.
  + Tipo: Char, que identificará o(s) serviço(s) complementar(es) (caso haja(m)) selecionados pelo cliente.
  + Duração: Int, que conterá a duração da prestação de serviço.
  + Custo: Int, que é derivado e representa o custo total.
  + Data: DATETIME, representa a hora e dia de início da atividade.
  + Avaliação: Float, que guarda, caso seja avaliado, a pontuação dada ao trabalho.
  + Pagamento: Char, que representa pagamento efetuado por Cartão de Crédito ou a dinheiro. Caso seja a dinheiro, o responsável por serviço é encarregado de fazer a cobrança.
  + Estado: Char, que identifica se o serviço está agendado, pendente ou terminado.
  + Observações: Texto, que poderá existir ou não, de acordo com os comentários que o cliente desejar colocar.
  + Morada: String, o local onde o trabalho é proposto ser efetuado. Não é dividido em Rua + Localidade pois a BD é pequena e o Parser do Bing Maps fará a devida compreensão. Implica menos necessidade de manutenção de tabelas a nível de implementação.
* Funcionário
  + Username: String, que poderá conter todo o tipo de caracteres. Com tamanho limitado a 10, por questões de eficiência.
  + Nome: String, que conterá o nome do funcionário.
  + Email: String, que conterá o email de registo do funcionário.
  + Password: String, que conterá a password do funcionário. Como a aplicação não necessita de elevada segurança, esta ficará guardada na mesma Base de Dados que todas as outras informações.
  + Turno: Char, que identifica o turno (manhã, tarde ou noite) em que o funcionário trabalha.
  + Rating: Float, que identifica o rating atual daquele funcionário.
  + Contacto: Int, que guarda o número do funcionário.
  + Avatar: FILESTREAM, que guardará a imagem do funcionário. É guardado diretamente na BD pois a aplicação é de pequena dimensão e diminui a carga de implementação. De facto, não há necessidade de ter um host de imagens e de fazer backups ao mesmo, ao invés de fazer apenas a toda a BD.
    1. Determinar chaves primárias, candidatas e alternativas

De forma a identifica unicamente cada uma das ocorrência das distintas entidades, foi necessário determinar os atributos chave.

Durante o processo de estudo, surgiram, para cala uma das entidades as seguintes chaves candidatas:

* Cliente: Username, Email
* Trabalho: Id
* Funcionário: Username, Email

Tendo sido selecionadas para chaves primárias:

* Cliente: Username
* Trabalho: Id
* Funcionário: Username

Optou-se pelo uso do username ao invés do email uma vez que é uma chave de menor comprimento.

* + 1. Desenho do diagrama ER

Apresentamos, de seguida, o desenho de diagrama E-R (Entidade-Relacionamento) de forma a representar conceptualmente as relações entre as entidades da base de dados.

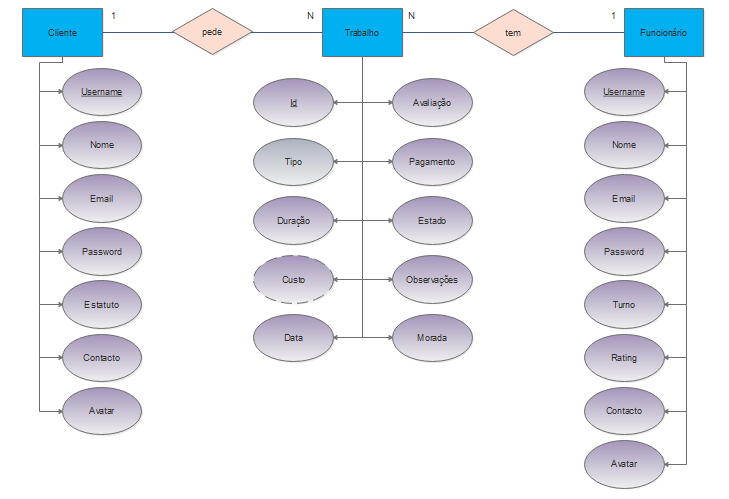


Figura 9 - Desenho do Diagrama ER

* + 1. Revisão do modelo de dados com o utilizador

Após a realização do modelo conceptual, este foi revisto pelo utilizador. Nenhum problema foi detetado e por isso o modelo de dados foi aceite.

1. Conclusões e Trabalho Futuro

A ideia de criar um serviço ao domicílio que funcionasse 24/365, deixando ao critério do grupo a escolha do tema e todo o trabalho envolvido na criação e fundamentação do mesmo, foi, sem dúvida, desafiante. Foi possível constatar que todo o esforço envolvido aquando do surgimento e definição de uma ideia exige tanto ou mais trabalho quanto a pô-la em prática. Nesta fase, apenas foi necessário elaborar um modelo geral da ideia a desenvolver, esclarecer alguns pontos e medidas de sucesso e viabilidade, e analisar os recursos necessários associados ao serviço. Ainda assim, sem uma boa fundamentação e sem o estabelecimento de metas, todos os passos futuros acabariam por levantar demasiadas dúvidas e poderiam acabar por divergir, tanto da ideia inicial, como entre os elementos do grupo.

Assim sendo, a equipa pretende, numa próxima fase, definir os requisitos do sistema, de forma a poder avançar com a arquitetura UML e, consequentemente, com o desenvolvimento da base de dados e da implementação da plataforma, quer ao nível do site, quer ao da aplicação móvel. O grupo espera conseguir finalizar as próximas etapas e tirar proveito das mesmas, levando consigo conhecimentos úteis para a vida profissional futura.

Lista de Siglas e Acrónimos

<<Apresentar uma lista com todas as siglas e acrónimos utilizados durante a realização do trabalho. O formato base para esta lista deverá ser da forma como abaixo se apresenta.>>

**BD** Base de Dados

DW Data Warehouse

OLTP *On-Line Analytical Processing*

*... ...*