

Atualização de Aplicação Web de Gestão de Processos de Formação

João Carlos Peixoto Freitas
(nrº 16968, regime diurno)

Orientação de
Joaquim G. P. Silva

LICENCIATURA EM ENGENHARIA EM SISTEMAS INFORMÁTICOS
ESCOLA SUPERIOR DE TECNOLOGIA
INSTITUTO POLITÉCNICO DO CÁVADO E DO AVE

Identificação do aluno

João Carlos Peixoto Freitas
Aluno número 16968, regime diurno
Licenciatura em Engenharia em Sistemas Informáticos

Orientação

Joaquim G. P. Silva

Informação sobre o Estágio

NKA - New Knowledge Advice, Lda
Travessa José Cruz, nº5, 4715-343 Braga, Portugal
Engenheiro João Magalhães

Resumo

A fim de manter o negócio competitivo no mercado, é preciso apostar em inovação. Por meio dessa iniciativa, a empresa consegue criar soluções eficientes, minimizar o impacto de eventualidades, antever situações adversas e gerar alternativas que proporcionem experiências positivas aos clientes.

Uma empresa que acredita que o seu produto já é bom o suficiente ao ponto de não precisar de inovação está destinada ao fracasso. Adiar ao máximo a atualização tecnológica, ou somente adotar medidas quando o mercado exige pode resultar numa perda de competitividade.

Tendo isto em consideração, com este trabalho foram atualizadas as tecnologias de um produto de forma a manter a sua operabilidade na tecnologia mais atual possível. Integraram-se no NkaAcademies, um produto desenvolvido pela NKA - *New Knowledge Advice*, atualizações ao nível de linguagem de programação e bibliotecas utilizadas de forma a tornar o produto o mais atual possível em termos tecnológicos e foram ainda desenvolvidas novas funcionalidades para adicionar futuramente ao produto.

Por fim, foi feita uma análise ao trabalho realizado, apresentando soluções desenvolvidas e tirando algumas conclusões sobre o projeto e o estágio.

Abstract

In order to keep the business competitive in the market, it is necessary to invest in innovation. With that in mind, the company is able to create efficient solutions, minimize the impact of eventualities, anticipate adverse situations and create alternatives that provide positive experiences for the customer.

A company that believes its product is good enough, that it doesn't need innovation is doomed to failure. Postponing the technological update as much as possible, or only adopting measures when the market demands it, can result in a loss of competitiveness.

Keeping this in mind, with this work the technologies of a product were updated in order to maintain its operability in the most current technology possible. In addition to NkaAcademies, a product developed by NKA - *New Knowledge Advice*, updates to the programming language and libraries used were made in order to make the product as up-to-date as possible in terms of technology, and the development of new functionalities to be added to the product in the future.

Finally, an analysis was made of the work accomplished, presenting developed solutions and some conclusions were taken about the project and the internship.

Agradecimentos

O desempenho no desenvolvimento deste projecto de estágio não teria sido o mesmo sem a ajuda e apoio de algumas pessoas, às quais gostaria de aqui expressar o meu reconhecimento.

Assim, gostaria de começar por agradecer à empresa NKA - New Knowledge Advice pela oportunidade de realizar estágio numa época complicada que vivemos e por toda a ajuda ao longo do estágio. Um obrigado em especial ao meu orientador na empresa, Eng. João Magalhães, pelo acompanhamento ao longo do meu trabalho e pela disposição de transmitir valores profissionais. Aproveito também para agradecer à restante equipa da NKA pela simpatia e familiaridade durante este percurso.

Gostaria de agradecer também ao meu orientador de estágio, o professor Joaquim Silva pela orientação, disponibilidade e conselhos dados ao longo desta etapa.

Por fim, agradeço à minha família e amigos por todo o apoio pois é também graças a eles que consegui acabar esta etapa da minha vida.

Conteúdo

1	Introdução	1
1.1	Objetivos	1
1.2	Contexto	1
1.3	Plano de trabalhos	2
1.4	Estrutura do documento	3
2	Processo de Reengenharia	5
2.1	Descrição das Tecnologias e Plataformas	5
2.2	Listagem de Funcionalidades do produto	6
2.3	Identificação das melhorias a realizar	9
2.4	Alterações no desenvolvimento da solução	9
2.4.1	Comparação de funções	9
2.4.2	<i>Improvement</i> da função <i>nka_mysql_result</i>	12
3	Desenvolvimento do novo módulo	13
3.1	Análise e Especificações	13
3.1.1	Requisitos Funcionais	13
3.1.2	Requisitos Não Funcionais	14
3.2	Arquitetura técnica	14
3.2.1	Diagrama de Atividades	15
3.2.2	Estrutura do código	16
4	Análise dos Resultados	19
4.1	Apresentação da solução desenvolvida	19
4.1.1	Solução desenvolvida - Processo de reengenharia	19
4.1.2	Solução desenvolvida - Desenvolvimento do novo módulo	20
4.2	Avaliação do nível de maturidade e cumprimento dos objetivos	24
4.3	Principais dificuldades encontradas	24

5	Conclusão Final	27
5.1	Principais conclusões do estágio	27
5.2	Pontos a melhorar/completar	27

Lista de Figuras

1.1	Diagrama de Gantt	2
2.1	Gestão da Formação	7
2.2	Painel Formador/Professor	8
3.1	Diagrama de Atividades	15
3.2	Estrutura do código	16
3.3	Interface Web - template teste	18
4.1	Menu Gestão Formação	20
4.2	Menu dados curso	20
4.3	Menu procurar	21
4.4	Menu procurar preenchido	22
4.5	Menu dados curso preenchido	22
4.6	Preenchimento automático	23
4.7	Menu preenchimento manual	23
4.8	Preenchimento manual	24

Lista de Tabelas

3.1	Requisitos funcionais	14
3.2	Requisitos não funcionais	14

Listagens de Código

2.1	Função <code>mysql_select_db</code>	10
2.2	Função <code>mysqli_select_db</code>	10
2.3	Função <code>mysql_result</code>	10
2.4	Função para substituir <code>mysql_result</code>	10
2.5	Função <code>mysql_field_name</code>	11
2.6	Função para substituir <code>mysql_field_name</code>	11
2.7	Improvement da função <code>nka_mysql_result</code>	12
3.1	Exemplo de request à base de dados	17
3.2	Função JavaScript utilizando <code>jQuery.ajax()</code>	17

Siglas & Acrónimos

HTML HyperText Markup Language. 17

MoSCoW Must Have, Should Have, Could Have, Won't Have. 3, 27

PHP HyperText Preprocessor. 3, 5, 6, 9, 12, 17, 19, 24, 25

SQL Structured Query Language. 10–12, 17

WYSIWYG *What You See Is What You Get*. 18

XAMPP Cross-Plataform(X), Apache(A), MariaDB(M), PHP (P) e Perl (P). 5

Glossário

Bootstrap *Framework* web para desenvolvimento de componentes de interface e front-end para sites e aplicações web usando HTML, CSS e JavaScript. 15, 17, 20

MoSCoW Método de priorização de funcionalidades ou requisitos. 13

1. Introdução

Acompanhar as novas tecnologias é fundamental para um negócio. É preciso investir em inovação para manter um negócio no mercado competitivo. A tecnologia, aliada a uma boa atualização, consegue proporcionar soluções eficientes às empresas.

Atualmente, no mundo em que vivemos, *updates* tecnológicos são lançados praticamente todos os dias relativos a sistemas de informação. É do maior interesse da parte de uma empresa, manter o seu produto atual e competitivo.

De momento, o produto NkaAcademies por parte da NKA - *New Knowledge Advice* encontra-se desatualizado a nível tecnológico, por se tratar de uma solução com algum tempo de desenvolvimento e permanência no mercado tecnológico como um forte produto a nível de gestão de processos de formação.

Como tal, este estágio visa principalmente a alteração das tecnologias utilizadas no produto NkaAcademies de forma a que a solução esteja preparada para futuras implementações desenvolvidas apenas em versões mais recentes e consiga assim alcançar todo o seu potencial para continuar a conquistar o mercado como um produto de excelência na gestão dos processos de formação.

1.1 Objetivos

Este estágio está diretamente relacionado com o produto NkaAcademies, que se trata de uma solução web completa para a gestão de processos de formação.

Pretende-se efetuar a migração completa do produto NkaAcademies para as versões mais recentes do PHP e do MySQLi e numa fase mais avançada do estágio, incorporar novas funcionalidades ao produto.

1.2 Contexto

Este projeto foi realizado no âmbito de estágio curricular da Licenciatura em Engenharia de Sistemas Informáticos, na empresa NKA - *New Knowledge Advice* em Braga.

O estágio curricular teve uma duração de aproximadamente quatro meses, tendo iniciado a 22 de fevereiro e terminado a 9 junho de 2021, de segunda a quinta feira num regime maioritariamente de teletrabalho, com exceção de algumas semanas em regime presencial.

A NKA - *New Knowledge Advice* é uma empresa tecnológica sediada em Braga, fundada em 2011 que concebe e desenvolve soluções globais vocacionadas para a otimização

de cada negócio, por meio de projetos específicos, fornecendo as soluções mais adequadas às crescentes exigências dos mercados. A empresa dedica-se ainda ao desenvolvimento de software, à implementação de sistemas e à consultoria e formação (NKA, 2021b).

1.3 Plano de trabalhos

No âmbito do estágio, foram identificadas as seguintes tarefas:

- Conhecer a empresa, tecnologias e projeto a desenvolver - Migração do produto NkaAcademies para as versões mais recentes do PHP e MySQLi;
- Investigação sobre PHP e MySQLi - Leitura de CHANGELOG das versões anteriores. Análise de funções *deprecated*.
- Alteração do código para a versão 8 PHP.
- Testes e correção de erros de compatibilidade
- Conhecer o projeto a desenvolver - módulo auxiliar ao processo de gestão de formação.
- Levantamento de requisitos e funcionalidades
- Desenvolvimento do módulo - Filtro de pesquisa e alteração manual e automática de documentos associados à gestão de formação.
- Testes e correção de erros
- Preparação da aplicação - Alteração de ligações para cloud. Limpeza de código.
- Atualização das alterações no servidor

Para planejar a execução das tarefas, foi elaborado um diagrama de Gantt de acordo com a Figura - 1.1.

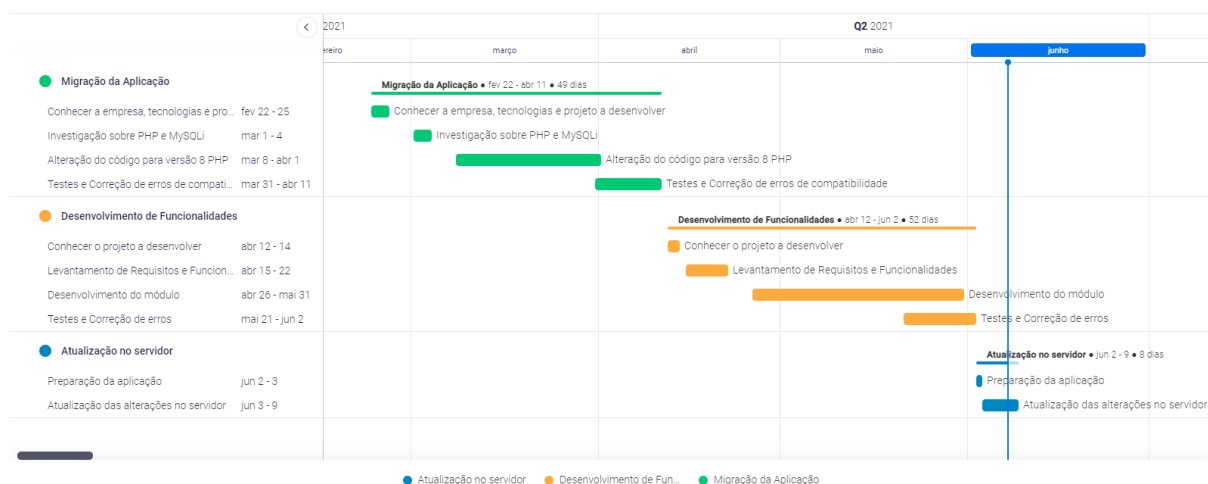


Figura 1.1: Diagrama de Gantt

1.4 Estrutura do documento

Este documento está dividido em vários capítulos e, cada um, dividido em subcapítulos, que explicam sucintamente os diferentes métodos de trabalho e tecnologias utilizadas.

Neste preciso capítulo, foram dadas a conhecer informações relativas ao estágio curricular e à empresa onde foi realizado, os objetivos propostos e a planificação das atividades com recurso ao diagrama de Gantt.

No segundo capítulo, Processo de reengenharia está descrito todo o processo de migração da aplicação NkaAcademies. Aqui serão descritas as tecnologias utilizadas ao longo do desenvolvimento do projeto, identificadas as principais falhas e melhorias a realizar e demonstradas algumas das principais diferenças entre as versões PHP 5.6 e 8.

No capítulo 3, Desenvolvimento do novo módulo foram dadas a conhecer todas as análises e especificações do sistema, ou seja, os requisitos funcionais e não funcionais categorizados pela metodologia MoSCoW. De seguida, foi apresentada a arquitetura do sistema e a estruturação do código de forma a facilitar a compreensão do sistema a desenvolver.

No capítulo 4 é feita uma análise a tudo o que foi desenvolvido durante o estágio curricular e feita uma observação em relação ao cumprimento dos objetivos inicialmente estipulados.

Por fim, segue-se o capítulo da Conclusão (5) onde é feita uma revisão geral ao estágio desenvolvido, bem como algumas reflexões.

2. Processo de Reengenharia

A área da reengenharia de software está ligada à reestruturação e redocumentação do software para o tornar mais fácil de perceber e alterar.

A funcionalidade do software não é alterada, e normalmente, não devem ser feitas grandes alterações a nível da arquitetura do sistema (Somerville, 2011).

Neste capítulo estão descritas as principais tecnologias e plataformas utilizadas no desenvolvimento quer da primeira e segunda parte do estágio. São ainda enunciadas algumas funcionalidades e capacidades do produto NkaAcademies que vai ser alvo de alteração tecnológica. Por fim, são descritas sucintamente as melhorias a realizar no produto e são demonstrados alguns exemplos das alterações realizadas, tais como um "frente a frente" de algumas funções suportadas nas diferentes versões do PHP e um *improvement* de funções.

2.1 Descrição das Tecnologias e Plataformas

Neste ponto serão descritas as tecnologias e plataformas utilizadas ao longo deste trabalho, tais como XAMPP, Laragon, Visual Studio Code.

Para a realização deste projeto foram utilizadas as plataformas e tecnologias requeridas pela empresa para facilitar a implementação e integração do projeto, e para tal foi necessário proceder à sua aprendizagem.

De seguida vão ser descritas de forma sucinta as tecnologias e plataformas utilizadas ao longo do projeto.

XAMPP

O XAMPP é formado por um pacote que inclui, base de dados MySQL, servidor web Apache e interpretadores para as linguagens de script. É essencialmente utilizado pelos desenvolvedores que pretendem criar um servidor web local no seu próprio computador, com a finalidade de realizar testes sem necessitar de acesso à rede (Friends, 2018).

Visual Studio Code

O VsCode é um editor de código-fonte simplificado com suporte para operações de desenvolvimento como *debugging*, execução de tarefas e controlo de versões (Microsoft, 2021).

Xdebug

O Xdebug é uma extensão para PHP que fornece uma variedade de recursos para melhorar a experiência de desenvolvimento de PHP (Rethans, 2020).

Laragon

O Laragon é uma maneira rápida e fácil de criar um ambiente de desenvolvimento isolado no Windows. Inclui Mysql, PHP Memcached, Redis, Apache (Keycdn, 2015).

2.2 Listagem de Funcionalidades do produto

NkaAcademies é uma solução web completa para a gestão de formação. A solução é composta por vários módulos e diferentes funcionalidades, que fazem parte também de diferentes perfis. Existe então implementado no produto do NkaAcademies uma plataforma completa para a gestão de formações, para os professores/formadores e para os alunos/formandos, cada uma com diferentes propósitos que culminam num produto bem estruturado e completo (NKA, 2021a).

Na **gestão da formação** (Figura 2.1), algumas funcionalidades incluem:

- Certificação de Diplomas
- Manuais Digitais
- Gestão de tarefas e projetos
- Faturação e gestão de recursos
- Cursos e-Learning
- Módulo de marketing
- Módulo de projetos

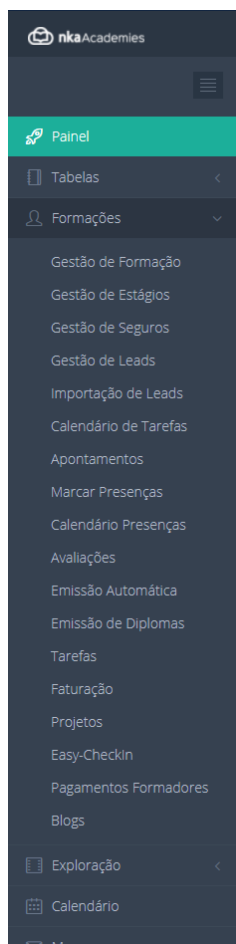


Figura 2.1: Gestão da Formação

O produto é ainda composto por uma plataforma para os seus **formadores** onde poderão encontrar entre outras, as seguintes funcionalidades (Figura 2.2):

- Presenças e avaliações online
- Listagem detalhada da turma
- Sistema interno de mensagens
- Partilhar apontamentos e exercícios
- Informação na "cloud" sincronizada
- Histórico de formações

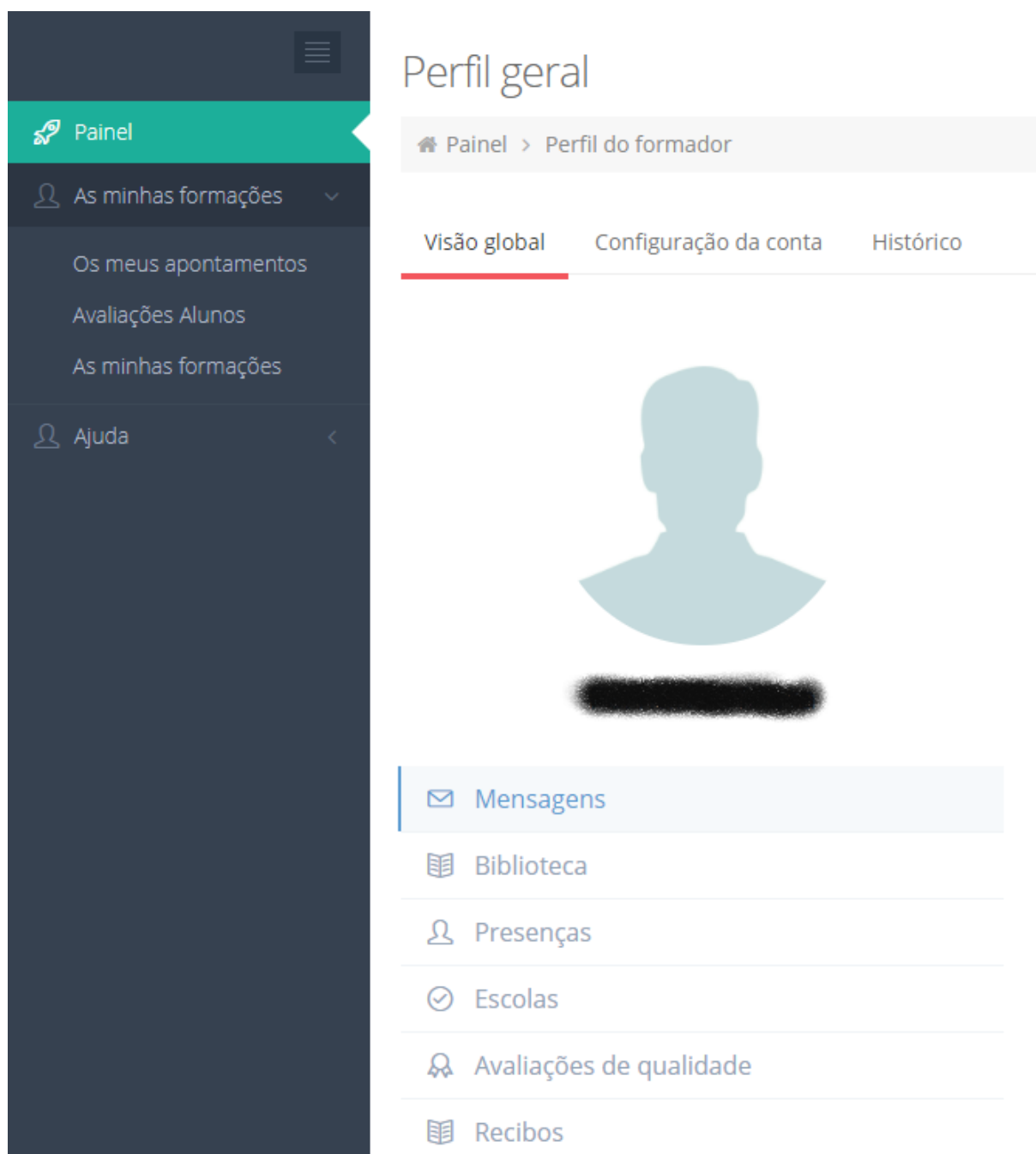


Figura 2.2: Painel Formador/Professor

A plataforma disponibiliza também para os seus **alunos** e formandos um local onde poderão encontrar entre outras, as seguintes funcionalidades:

- Partilha de diploma
- Descarregar manuais e documentos
- Devolver feedback
- Histórico de formação
- Manter o contacto com antigos colegas de formação
- Detalhes da formação

2.3 Identificação das melhorias a realizar

Atualmente, no mundo em que vivemos, *updates* tecnológicos são lançados praticamente todos os dias relativos a sistemas de informação. É do maior interesse da parte de uma empresa, manter o seu produto atual e competitivo.

De momento, o produto NkaAcademies encontra-se desatualizado, utilizando versões antigas de PHP(5.6) e MySQL. Foi então proposto desenvolver a migração de todo o produto NkaAcademies para as versões mais recentes, suportadas no *host* utilizado pela empresa, do PHP(5.6 -> 8.0.2) e do MySQL com o objetivo de atualizar o produto e também, em casos futuros o uso de novas funcionalidades que surgiram apenas em versões superiores à que o produto estava desenvolvido.

As melhorias a realizar passaram maioritariamente pela substituição total da biblioteca PHP MySQL para MySQLi, uma vez que grande parte das funções que fazem parte da biblioteca MySQL foram descontinuadas / *deprecated*, enquanto noutros casos apenas foram feitas alterações ao nível da chamada da função ou dos parâmetros (e ordem) recebidos. Posto isto, procedeu-se à sua alteração total no produto NkaAcademies, sendo que no caso de funções *deprecated* foram implementadas alternativas *hard-code* em PHP.

Uma vez que se trata de uma migração completa de um produto pré-existente o objetivo principal definido para o cumprimento desta fase tem a ver com a compatibilidade das alterações feitas. As novas funções e todas as modificações realizadas ao nível do produto devem ser implementadas tendo em vista a sua compatibilidade com as versões anteriores, que neste caso são as versões PHP 5.6 e 7 e a compatibilidade com outras funcionalidades já existentes no produto garantindo a operabilidade e o bom funcionamento da aplicação na nova versão tecnológica.

2.4 Alterações no desenvolvimento da solução

Neste capítulo serão apresentadas algumas das principais alterações implementadas no produto NkaAcademies na fase da migração da aplicação para as versões mais recentes do PHP e MySQL.

As principais alterações efetuadas no projecto NkaAcademies foram ao nível da chamadas de funções, implementação de novos métodos devido à descontinuidade de funções da biblioteca MySQL, parâmetros (e ordem) recebidos por funções e correções de erros de compatibilidade entre versões.

2.4.1 Comparação de funções

Neste sub-capítulo serão comparadas as funções das bibliotecas MySQL e MySQLi, em relação a *inputs* e *outputs* de algumas das funções utilizadas.

mysql_select_db vs mysqli_select_db

Neste caso, será apresentado um excerto de código relativo a uma função que foi atualizada na biblioteca MySQLi, recebendo o mesmo tipo de parâmetros mas que trocou

a ordem de como eram incorporados na função.

A função *mysql_select_db* (listagem 2.1) seleciona uma base de dados do tipo MySQL.

```
function mysql_select_db($database , $link_identifier=NULL):bool
```

Listagem 2.1: Função *mysql_select_db*.

A função da Listagem 2.1 recebe como argumentos:

- **\$database**: nome da base de dados que deverá ser selecionada.
- **\$link_identifier**: conexão ao MySQL.

A função *mysqli_select_db* (listagem 2.2) seleciona a base de dados a ser utilizada ao realizar *queries* do tipo MySQLi.

```
function mysqli_select_db($mysql , $database):bool
```

Listagem 2.2: Função *mysqli_select_db*.

A função da Listagem 2.2 recebe como parâmetros:

- **\$mysql**: *connection string* de acesso à base de dados.
- **\$database**: nome da base de dados que deverá ser selecionada.

mysql_result vs nka_mysqli_result

Neste exemplo será apresentado nos excertos de código a alteração que teve de ser feita quando a função da biblioteca MySQL não foi continuada na biblioteca MySQLi, fazendo com que tivesse de ser criada uma equivalente.

A função *mysql_result* (listagem 2.3) obtém os dados de um determinado resultado.

```
function mysql_result($result , $row , $field=0):string
```

Listagem 2.3: Função *mysql_result*.

A função da Listagem 2.3 recebe como argumentos:

- **\$result**: resultado da *query* SQL a ser executado.
- **\$row**: número da linha do resultado.
- **\$field**: nome do campo a ser procurado (tabela).

A função *nka_mysqli_result* (listagem 2.4), utilizada para substituir a função *deprecated mysql_result* foi implementada recebendo os mesmos argumentos que a anterior, e retornando uma *string*.

```
if (!function_exists('nka_mysqli_result')) {
    function nka_mysqli_result($res , $row , $field=0) {
        $res->data_seek($row);
        $datarow = $res->fetch_array();
```

```

        return $datarow[ $field ];
    }
}

```

Listagem 2.4: Função para substituir `mysql_result`.

A função da Listagem 2.4 recebe como argumentos:

- **\$result**: resultado da *query* SQL a ser executado.
- **\$row**: número da linha do resultado.
- **\$field**: nome do campo a ser procurado (tabela).

mysql_field_name vs mysqli_field_name

Nas Listagens 2.5 e 2.6 está mais uma vez demonstrado o caso em que uma função da biblioteca MySQL é descontinuada para MySQLi e teve de se proceder à sua implementação.

A função *mysql_field_name* (listagem 2.5) obtém o nome do campo especificado num resultado.

```
function mysql_field_name($result , $field_offset):string|false
```

Listagem 2.5: Função `mysql_field_name`.

A função da Listagem 2.5 recebe como parâmetros:

- **\$result**: resultado da *query* SQL a ser executada.
- **\$field_offset**: *index/offset* de um campo numérico.

A função *mysqli_field_name* (listagem 2.6), utilizada para substituir a função *deprecated mysql_field_name* foi implementada no projeto recebendo os mesmos parâmetros que a anterior, e retornando uma *string* ou *false*.

```

if (!function_exists('mysqli_field_name')) {
    function mysqli_field_name($result , $field_offset){
        $properties = mysqli_fetch_field_direct($result , $field_offset);
        return is_object($properties) ? $properties->name : false;
    }
}

```

Listagem 2.6: Função para substituir `mysql_field_name`.

A função da Listagem 2.6 recebe como parâmetros:

- **\$result**: resultado da *query* SQL a ser executada.
- **\$field_offset**: *index/offset* de um campo numérico.

2.4.2 *Improvement* da função `nka_mysql_result`

Neste caso está demonstrado um exemplo *improvement* de uma função que já tinha sido desenvolvida para substituir uma função descontinuada da biblioteca MySQL.

A função `nka_mysql_result_v2` (listagem 2.7), foi um *improvement* da função previamente implementada `nka_mysql_result`. Esta alteração deveu-se ao facto de alguns erros na incompatibilidade das diferentes versões(7 e 8) do PHP em que o `$datarow` chegava *empty*, enquanto que na versão PHP5.6 a mesma função não apresentava erro. Verificou-se que a mesma função se comportava de forma diferente de versão para versão e como tal, procedeu-se à sua remodelação passando agora a retornar *string* ou *null* resolvendo o problema.

```

if (!function_exists( 'nka_mysql_result_v2' )) {
    function nka_mysql_result_v2( $res , $row , $field=0) {
        $res->data_seek( $row );
        $datarow = $res->fetch_array ( );
        $retval = null;
        $cond = ! empty( $datarow );
        if( $cond ){
            $retval = $datarow[ $field ];
        }
        return $retval;
    }
}

```

Listagem 2.7: *Improvement* da função `nka_mysql_result`.

A função da Listagem 2.7 recebe como argumentos:

- **\$result**: resultado da *query* SQL a ser executado.
- **\$row**: número da linha do resultado.
- **\$field**: nome do campo a ser procurado (tabela).

3. Desenvolvimento do novo módulo

Este capítulo apresenta o trabalho desenvolvido numa segunda parte do estágio, o desenvolvimento de um módulo a implementar no produto NkaAcademies. Foi necessário analisar requisitos, tratar da modelação do produto e da sua implementação.

3.1 Análise e Especificações

Na identificação de requisitos são enunciadas todas as funcionalidades, de forma enumerada, que se pretendem implementar no *software*. Os diferentes requisitos, funcionais e não funcionais, encontram-se classificados pela metodologia *MoSCoW* que identifica os requisitos de elevada prioridade até aos menos prioritários.

Cada tipo de requisito classificado (*Must*, *Should*, *Could*, e *Would*) representam diferentes significados face ao foco e importância que deverá ser levada em conta durante o processo de desenvolvimento.

Assim sendo:

1. ***Must***: classificados como requisitos mais críticos ou indispensáveis para o produto, pretende-se o total foco sobre este tipo de requisitos durante o processo de desenvolvimento;
2. ***Should***: são tão importantes como os requisitos classificados como *Must Have*, contudo não são considerados tão priorizados;
3. ***Could***: entende-se como requisitos desejáveis, mas também não são necessários;
4. ***Would***: estes são os requisitos menos críticos e com menor valor, podendo ser implementados ou não.

3.1.1 Requisitos Funcionais

Os requisitos funcionais dizem respeito a todas as funcionalidades ou pressupostos de como o sistema deve reagir a entradas específicas e como se deverá comportar. Estes tipos de requisitos podem-se entender como funcionalidades que o utilizador pode interagir de forma direta e mais visível.

A Tabela 3.1 - enuncia os requisitos funcionais estipulados para o sistema.

#	Requisito	Descrição	Prioridade
RF01	Pesquisar formação	O sistema tem que permitir ao utilizador a procura pelo código da formação ou código de curso	MUST
RF02	Escolher declaração	O sistema tem que permitir ao utilizador a escolha da declaração que pretende imprimir / preencher.	MUST
RF03	Editar manualmente	O sistema deverá permitir ao utilizador a edição manual dos campos a preencher	MUST
RF04	Criar campos	O sistema deve permitir a criação de novos campos a serem adicionados às declarações (<i>template</i>)	COULD
RF05	Editar automaticamente	O sistema deverá permitir ao utilizador a edição automática dos campos a preencher, conforme os resultados da procura	MUST
RF06	Imprimir documento	O sistema deverá permitir ao utilizador a impressão dos documentos editados.	SHOULD

Tabela 3.1: Requisitos funcionais

3.1.2 Requisitos Não Funcionais

Requisitos não funcionais são as propriedades comportamentais relacionadas ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas. Estes requisitos dizem respeito a como as funcionalidades serão entregues ao utilizador do software.

A Tabela 3.2 - encontram-se especificados os requisitos não funcionais do sistema.

#	Descrição	Categoria	Prioridade
RNF01	As novas funções deverão ser implementadas de forma a que sejam compatíveis com outras funcionalidades já existentes	Compatibilidade	MUST
RNF02	O módulo deverá ser implementado em JavaScript, PHP com acesso à base de dados MySQL	Operacional	MUST
RNF03	Apenas utilizadores autenticados e com permissão devem ter acesso módulo desenvolvido	Segurança	MUST

Tabela 3.2: Requisitos não funcionais

3.2 Arquitetura técnica

As tecnologias e linguagens utilizadas para o desenvolvimento deste módulo a adicionar ao produto NkaAcademies foram essencialmente descritas no Capítulo 2.1, com adição de

JavaScript e Bootstrap para desenvolver a componente web da aplicação.

3.2.1 Diagrama de Atividades

Para descrever o processo de negócio foi utilizado o diagrama de atividade, uma vez que permite modelar o comportamento do sistema incluindo a sequência e as condições de execução de ações.

A Figura 3.1 demonstra o diagrama de atividades desenvolvido para o sistema.

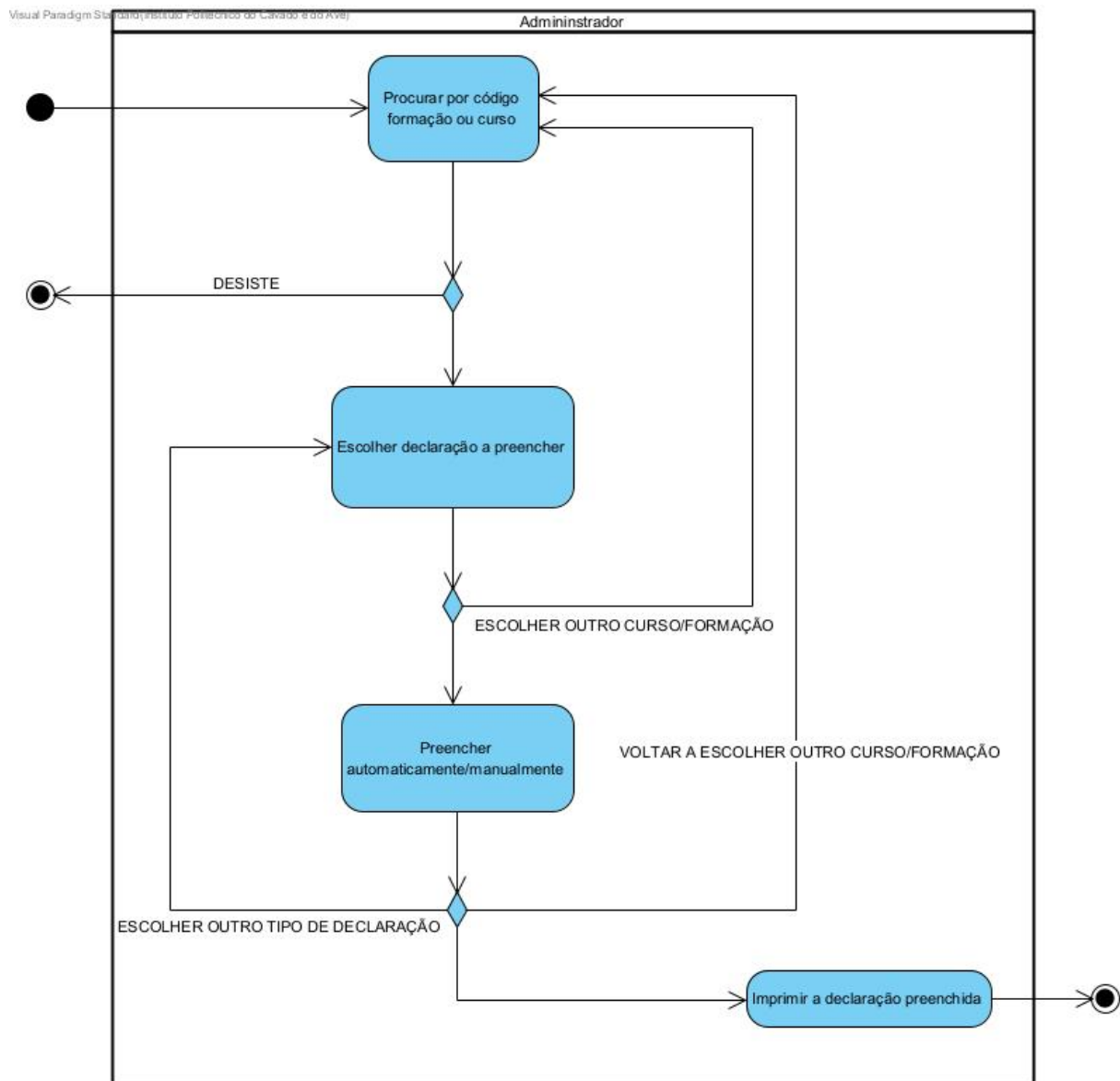


Figura 3.1: Diagrama de Atividades

Recorrendo à visualização do diagrama da Figura 3.1, é possível perceber os diferentes passos que correspondem à atividade geral do desenvolvimento desta fase do projeto, o preenchimento de documentos de apoio aos processos de gestão de formação. As atividades foram então: filtrar a procura a um resultado, escolher um *template* / declaração

predefinido, proceder ao seu preenchimento manual/automático e posteriormente à sua impressão caso seja pretendido.

No desenho deste diagrama está subentendido que o cliente está devidamente autenticado e verificadas as suas permissões no acesso a este tipo de ferramenta dentro da aplicação.

3.2.2 Estrutura do código

Neste sub-capítulo será apresentada a estrutura do código desenvolvido, bem como alguns excertos de código para exemplificar as diferentes camadas existentes no projeto: base de dados, *back-end* e *front-end*.

A Figura 3.2 apresenta um esquema gráfico, de forma a que se perceba melhor esta estrutura e as relações entre camadas.

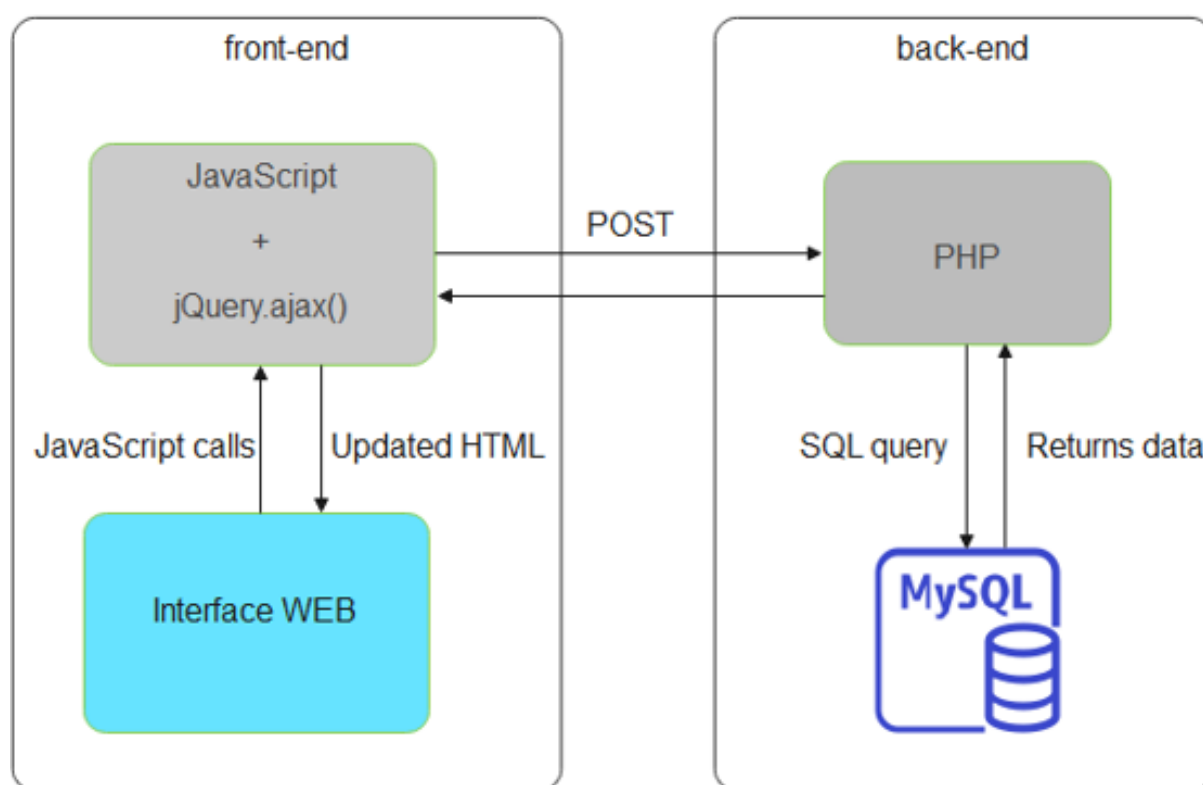


Figura 3.2: Estrutura do código

Base de Dados

Quanto à base de dados utilizada para o desenvolvimento do projeto, não foi feito qualquer tipo de modelação uma vez que foi reaproveitada a base de dados já existente do produto NkaAcademies tendo sido apenas necessário acrescentar alguns dados em tabelas(*templates*) já existentes a fim de testes e futura implementação.

back-end

A parte do *back-end* da aplicação foi desenvolvida em PHP. O principal objetivo do trabalho realizado no *back-end* diz respeito a interações com a base de dados, ou seja, tudo que envolva *queriesSQL* e ligações a bases de dados são definidas neste sector.

Para uma melhor compreensão, será apresentada na Listagem 3.1 um exemplo de uma interação com a base de dados onde basicamente é retornado o conteúdo do *template* selecionado pelo utilizador (*\$_POST["diploma"]*). Caso se verifique que o *iduser* / cliente existe e o *template* selecionado está definido na base de dados, é devolvido o seu conteúdo para a interface web.

```
case 2073:

$result = mysqli_query($con, "SELECT_*_from_XXX.templates
where_diploma='". $_POST["diploma"] . "' and_iduser='$varemp'");

if (mysqli_num_rows($result)!=0)
{
echo str_replace("table_table-bordered",
" ",utf8_encode(nka_mysql_result($result,0,"body")));
}
else
{
echo "Sem_Template_Definido";
}

mysqli_close($con);

break;
```

Listagem 3.1: Exemplo de request à base de dados

front-end

A parte do *front-end* foi desenvolvida em HTML, recorrendo a *modals* do Bootstrap para implementar a interface web e JavaScript que contém as funções que são o elo de ligação entre a interface web e o *back-end*, que permite atualizar e tornar dinâmica a parte gráfica/visual do sistema.

A Listagem 3.2 demonstra a implementação de uma função JavaScript que complementa a ação realizada ao *back-end* na listagem 3.1.

De salientar ainda, a importância do uso de *jQuery.ajax()* que permite utilizar JavaScript para enviar *asynchronous http request* e obter resposta em vários formatos, e também para atualizar partes de uma página web (que use JavaScript) sem ter que recarregar a página web completa (TutorialsTeacher, 2020).

Assim sendo, o *jquery* neste caso é responsável por passar o parâmetro **diploma** para o *back-end* de forma a ser processada a querySQL correspondente à declaração que o cliente selecionou.

```
function carregatexto(){
```

```

var resultado = $[ 'ajax ' ] ( {
  type: 'POST',
  url: 'phpScripts/dbload.php',
  data: 'kind=2073&diploma='
    +document.getElementById("template").value ,
  async: false } ) [ 'responseText ' ];

  $( ".summernote" ).summernote( "code", resultado );

  initcampos ();
}

```

Listagem 3.2: Função JavaScript utilizando jQuery.ajax()

Na Figura 3.3 está representada a interface web resultante das listagens de código descritas acima nos pontos 3.1 e 3.2.

De salientar o uso do *summernote* que é uma biblioteca do JavaScript que ajuda na criação de um editor WYSIWYG online (SummernoteTeam, 2020).

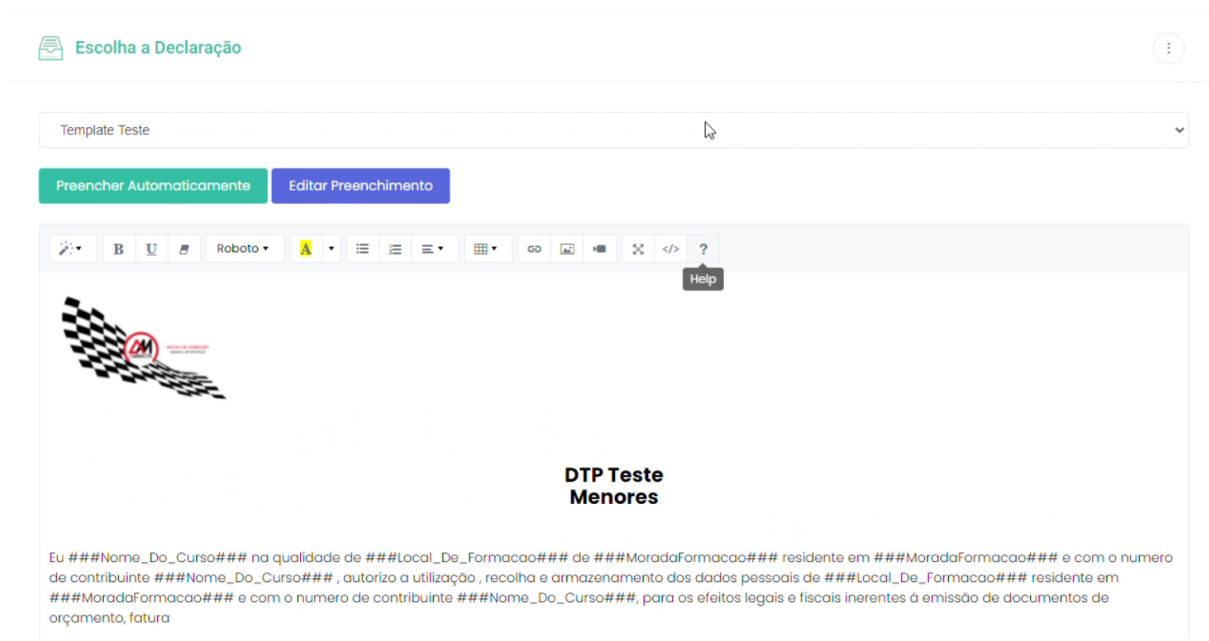


Figura 3.3: Interface Web - template teste

4. Análise dos Resultados

Neste capítulo encontra-se uma revisão geral ao que foi abordado, através de uma apresentação de resultados obtidos, uma avaliação do nível de maturidade e cumprimentos dos objetivos bem como de algumas dificuldades ultrapassadas no desenvolvimento do projeto.

4.1 Apresentação da solução desenvolvida

Nesta secção serão apresentadas as soluções desenvolvidas divididas em dois sub-capítulos referentes às duas partes que compuseram o estágio, o processo de reengenharia e o desenvolvimento do novo módulo.

4.1.1 Solução desenvolvida - Processo de reengenharia

A primeira parte do projeto de estágio prende-se no processo de reengenharia, descrito no capítulo 2, que teve como principal objetivo a migração completa do produto NkaAcademies para as versões mais recentes do PHP e MySQL.

Este tipo de alteração torna-se um pouco complicado de demonstrar, mas a migração total da solução NkaAcademies foi realizada e está neste momento, aos poucos, a ser introduzida no servidor *host* responsável pelo armazenamento do produto para não causar conflitos, uma vez que se pretende garantir o contínuo bom funcionamento da solução *online*.

Segue na Figura 4.1 uma demonstração do painel "Gestão de Formação" funcional de acordo com as novas atualizações tecnológicas.

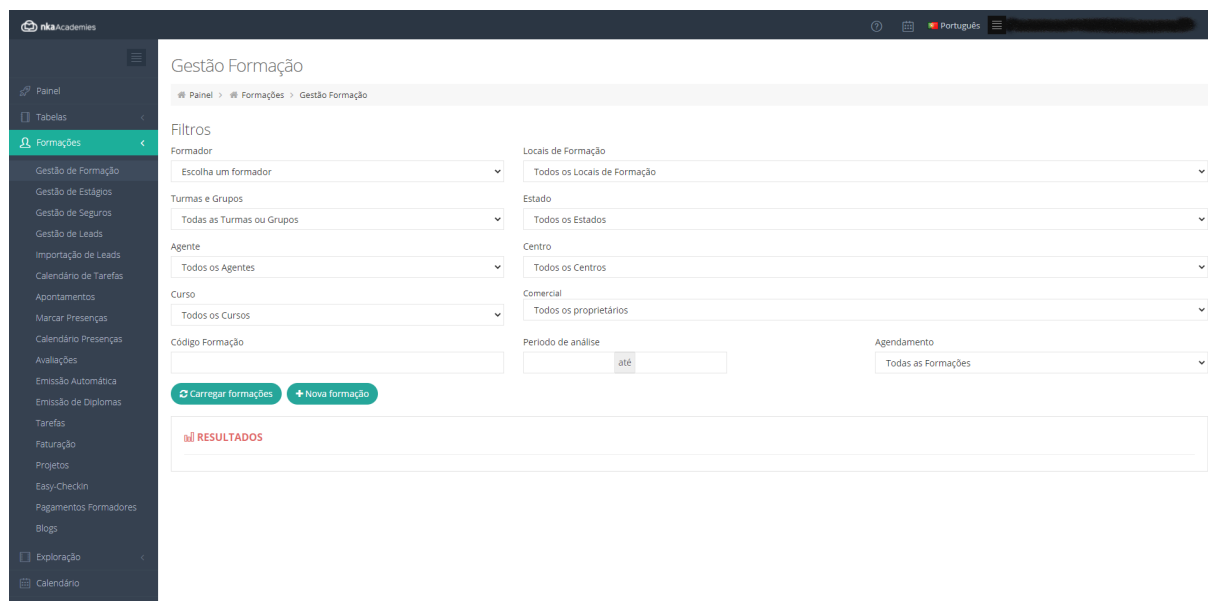


Figura 4.1: Menu Gestão Formação

4.1.2 Solução desenvolvida - Desenvolvimento do novo módulo

A segunda parte do projeto de estágio estava direcionada para a implementação de um módulo / funcionalidade a adicionar ao produto NkaAcademies, um sistema de apoio à gestão de processos de formação - preenchimento de dossiers pedagógicos, de forma a passar a customização e a parte de gerar estes dossiers para o lado do cliente.

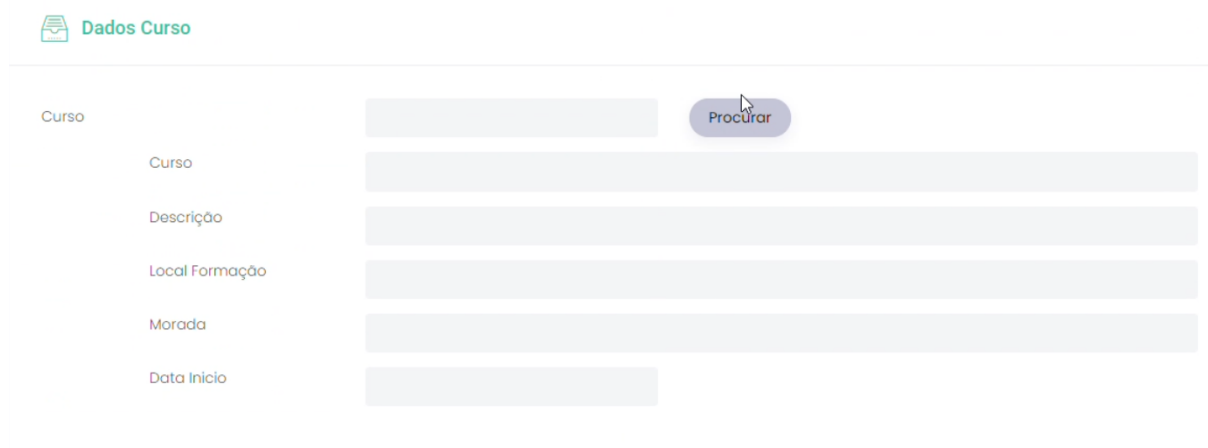
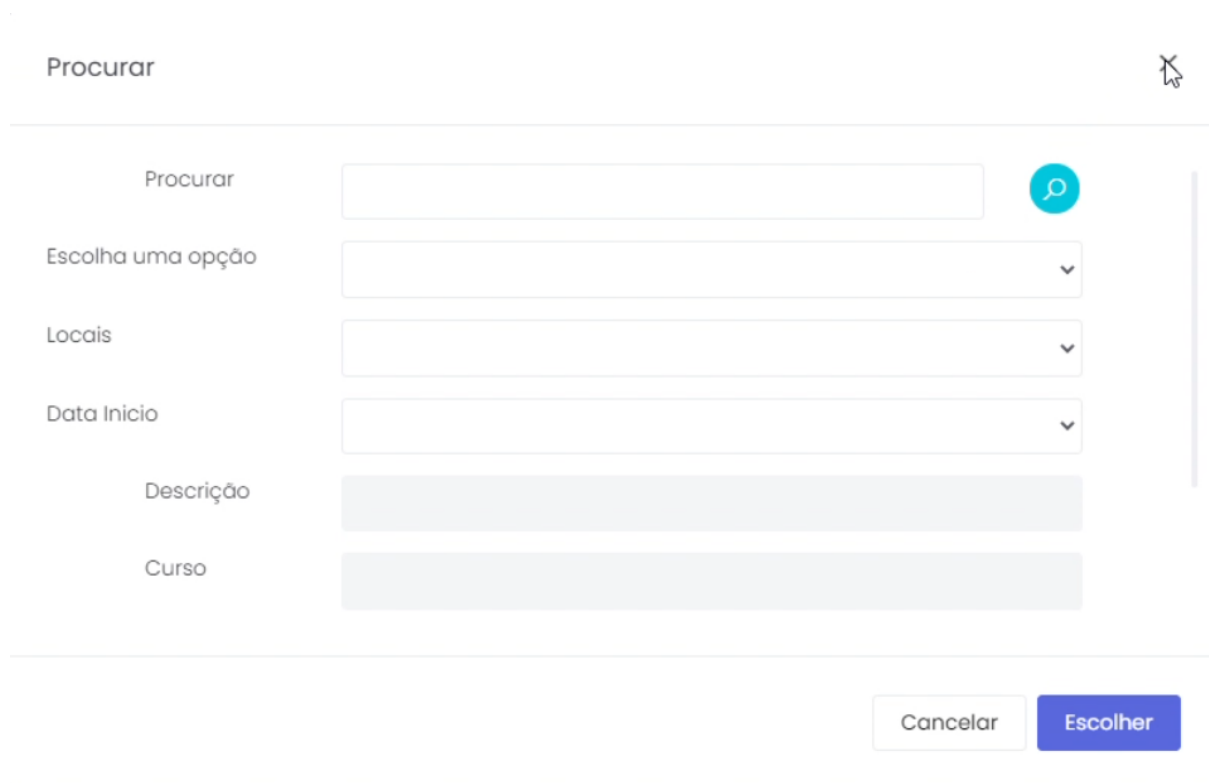


Figura 4.2: Menu dados curso

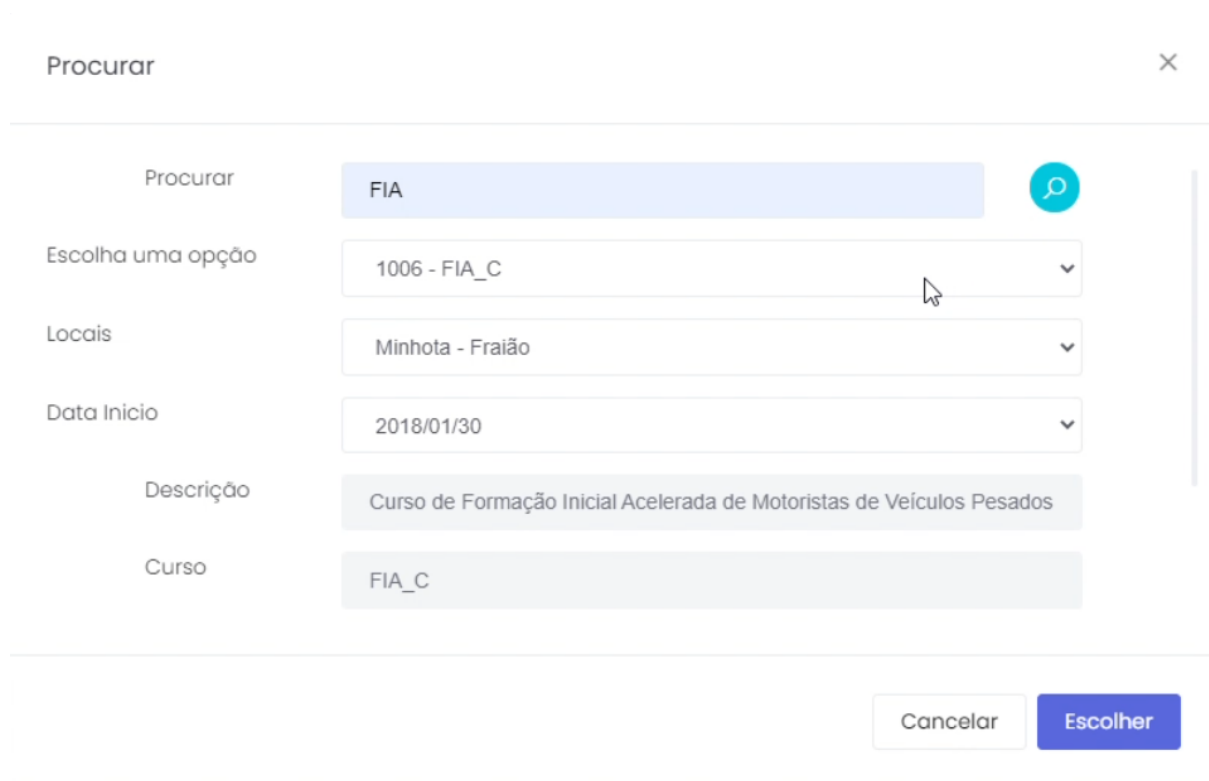
Deste modo, o cliente começaria com algo como a Figura 4.2, onde ao clicar no botão **Procurar** seria lançado um *pop-up*, que na verdade é apenas a utilização de *modals* do Bootstrap como é possível observar na Figura 4.3.



The image shows a web interface titled "Procurar" (Search). It features a search bar with a magnifying glass icon. Below the search bar are several filter options, each with a dropdown arrow: "Escolha uma opção", "Locais", and "Data Inicio". There are also two text input fields labeled "Descrição" and "Curso". At the bottom right, there are two buttons: "Cancelar" (Cancel) and "Escolher" (Choose).

Figura 4.3: Menu procurar

Após ser aberto o "Menu Procurar", o cliente deverá pesquisar pelo código da formação ou pelo nome do curso para que a *drop-down box* "Escolha uma opção" seja preenchida com os resultados obtidos da pesquisa realizada. Conforme altera a opção obtida na *drop-down box*, todos os campos são dinâmicos e deverão se ajustar à informação que está a ser selecionada, como é possível visualizar na Figura 4.4.



Procurar	
Procurar	FIA
Escolha uma opção	1006 - FIA_C
Locais	Minhota - Fraião
Data Inicio	2018/01/30
Descrição	Curso de Formação Inicial Acelerada de Motoristas de Veículos Pesados
Curso	FIA_C
<div>Cancelar Escolher</div>	

Figura 4.4: Menu procurar preenchido

Uma vez seleccionados os campos que pretende visualizar, e seleccionando o botão "Escolher", os dados (Figura 4.5) são automaticamente carregados para fora do *modal*, sendo preenchidos nos locais anteriormente vazios na Figura 4.2.



Dados Curso	
Curso	1032
Curso	FEA_C
Descrição	Formação Esp. Acelerada para Motoristas de Veículos Pesados de Mercadorias -CAM 35h DMat
Local Formação	Minhota - Fraião
Morada	Av. Robert Smith 41,
Data Inicio	2018/05/29
<div>Procurar</div>	

Figura 4.5: Menu dados curso preenchido

Depois de filtrados os resultados apenas a um, resta ao cliente seleccionar o tipo de *template* que deseja observar, que neste caso apenas está demonstrado um para efeitos de teste (Figura 3.3). Se optar por "Preencher Automaticamente", como ilustrado na Figura 4.6 os campos serão automaticamente substituídos pelos dados recolhidos acima, na Figura 4.5.

The screenshot shows a web interface for 'Template Teste'. At the top, there are two buttons: 'Preencher Automaticamente' (highlighted in green) and 'Editar Preenchimento'. Below these is a rich text editor toolbar with icons for bold, italic, underline, text color, background color, bulleted list, numbered list, link, unlink, and help. The main content area displays a logo on the left and the text 'DTP Teste Menores' in the center. Below this, there is a paragraph of text: 'Eu FEA_C na qualidade de Minhota - Friaão de Av. Robert Smith 41, residente em Av. Robert Smith 41, e com o numero de contribuinte FEA_C, autorizo a utilização, recolha e armazenamento dos dados pessoais de Minhota - Friaão residente em Av. Robert Smith 41, e com o numero de contribuinte FEA_C, para os efeitos legais e fiscais inerentes à emissão de documentos de orçamento, fatura'. At the bottom left, it says 'Assinatura:'. On the right side, there are three green notification boxes, each with a checkmark and the text 'Documento preenchido automaticamente!' or 'Documento atualizado!'.

Figura 4.6: Preenchimento automático

Caso o cliente opte por fazer o preenchimento das declarações manualmente, também é possível, como observado na Figura 4.7 apenas precisando de alterar para a informação que pretende e "Preencher".

The screenshot shows a 'Preenchimento Manual' dialog box with a close button (X) in the top right corner. The dialog contains a list of form fields for manual entry. The first three fields are pre-filled: 'Nome_Do_Curso' with 'FEA', 'Local_De_Formacao' with 'Minhota - Fajacal', and 'MoradaFormacao' with 'Av. Robert Smith 41,'. The remaining fields are empty: another 'MoradaFormacao', 'Nome_Do_Curso', 'Local_De_Formacao', 'MoradaFormacao', and 'Nome_Do_Curso'. At the bottom right, there are two buttons: 'Cancelar' and 'Preencher' (highlighted in blue).

Figura 4.7: Menu preenchimento manual

Por fim, as alterações indicadas na Figura 4.7 são feitas e o resultado da declaração é atualizado, como é possível observar na Figura 4.8.

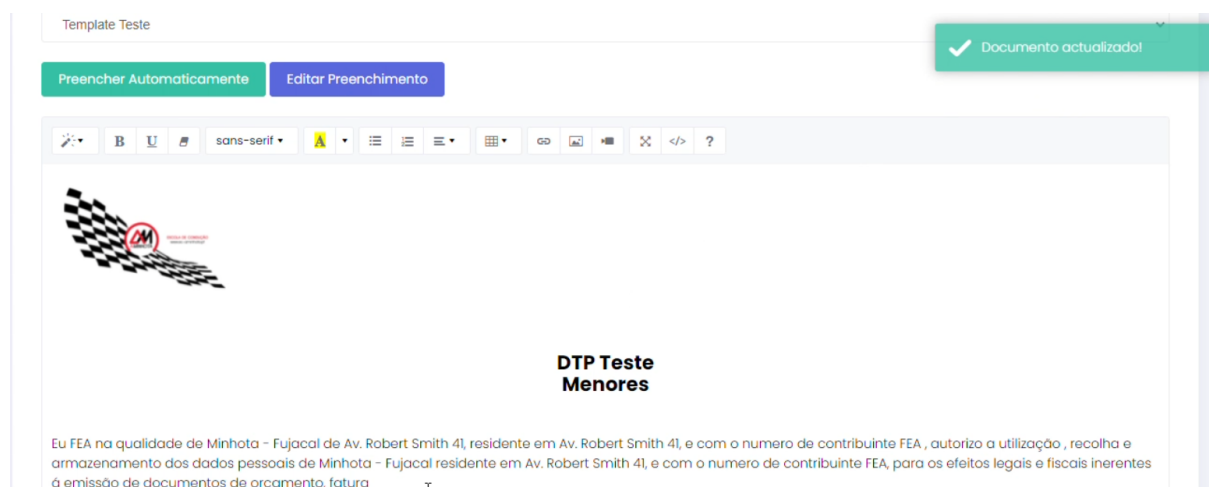


Figura 4.8: Preenchimento manual

4.2 Avaliação do nível de maturidade e cumprimento dos objetivos

Após a apresentação da solução desenvolvida (sub-capítulo 4.1) foi elaborada uma análise de comparação entre o que estava inicialmente estipulado para o estágio e o que foi desenvolvido. No geral, os objetivos e requisitos definidos foram cumpridos na sua totalidade.

Durante o todo o processo de estágio houve necessidade de fazer investigação sobre as principais alterações de cada versão da linguagem PHP, bem como da forma como as tecnologias operam, uma vez que não detinha os principais conhecimentos sobre as linguagens utilizadas. Como tal, o desenvolvimento deste projeto de estágio exigiu uma elevada auto motivação para o cumprimento dos objetivos propostos.

4.3 Principais dificuldades encontradas

Durante a migração da aplicação entre as versões do PHP 5.6 -> 8, mesmo apesar da investigação realizada a fim de perceber as principais mudanças entre versões, as funções e bibliotecas *deprecated* / descontinuadas houveram bastantes problemas em conseguir uma implementação de forma a garantir a compatibilidade e operabilidade do produto nas versões mais recentes.

Trata-se de uma linguagem de programação que sofreu algumas alterações desde a versão em que estava implementada no produto, e acima de tudo de uma versão recente da tecnologia que nem um ano possui, ou seja, mesmo a nível de documentação e *CHANGELOG* muito pouco evoluído.

Outra das adversidades que pode ser considerada um entrave ao projeto inicialmente foi o facto de estar a trabalhar com tecnologias novas em termos de conhecimento próprio, como o PHP e JavaScript tendo sido necessário proceder à sua aprendizagem.

5. Conclusão Final

5.1 Principais conclusões do estágio

Neste período de quatro meses de estágio, o contacto com novas tecnologias, a integração numa empresa e a experiência de desenvolver projectos inseridos em ambientes de grande escala são algumas das competências adquiridas que completam a formação académica.

O estágio ofereceu-me uma experiência inicial muito importante no desenvolvimento de competências úteis para o meu futuro profissional, uma vez que desde muito cedo fui parte integrante em algumas reuniões de equipa de projetos, ou seja, experienciava de perto a maneira de como um produto era trabalhado e organizado até ao seu desenvolvimento e foram-me transmitidos muitos valores profissionais dentro da empresa.

Por fim, elaborando uma análise global ao trabalho desenvolvido é visível todo o trabalho realizado ao longo destes quatro meses. O objetivo, na sua totalidade, ainda se encontra um pouco longe do atingido, contudo o objetivo delineado para o estágio curricular foi cumprido uma vez que os requisitos impostos no início do estágio foram concluídos, embora a última parte do desenvolvimento do módulo de apoio à gestão dos processos de formação - dossiers pedagógicos continue em fase de protótipo e como futura implementação no produto NkaAcademies.

5.2 Pontos a melhorar/completar

Neste projeto, os pontos a melhorar estão mais direcionados ao desenvolvimento da ferramenta de apoio ao preenchimento de documentos associados aos processos de formação, ou seja, à segunda parte do estágio.

O módulo desenvolvido ainda está em fase protótipo e como potencial implementação futura no produto NkaAcademies, e sendo que existiram alguns requisitos estipulados, de menor prioridade de acordo com a metodologia MoSCoW que não foram implementados devido à falta de tempo considera-se que estes são alguns dos pontos que podem ser completados futuramente.

Bibliografia

- Friends, A. 2018. *About the xampp project*. https://www.apachefriends.org/pt_br/about.html.
- Keycdn. 2015. *100+ awesome web development tools and resources*. <https://www.keycdn.com/blog/web-development-tools>.
- Microsoft. 2021. *Vscode faq*. <https://code.visualstudio.com/docs/supporting/FAQ>.
- NKA. 2021a. *O que é a nkaacademies?* <https://www.nkaacademies.pt/pt/#what-we-do>.
- NKA. 2021b. *Quem somos*. <https://nka.pt/pt-pt/content/quem-somos>.
- Rethans, Derick. 2020. *Xdebug - debugger and profiler tool for php*. <https://xdebug.org/>.
- Somerville, I. 2011. *Software engineering* v. 264–274. Addison-Wesley 9th edn.
- SummernoteTeam. 2020. *Summernote*. <https://summernote.org/>.
- TutorialsTeacher. 2020. *jquery ajax introduction*. <https://www.tutorialsteacher.com/jquery/jquery-ajax-introduction>.