

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

Não é segredo para ninguém que, 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 ao consumidor.

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.

Abstract

It is no secret that, 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.

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	1
1.4	Estrutura do documento	2
2	Processo de Reengenharia	5
2.1	Sistemas AS-IS	5
2.1.1	Descrição da Arquitetura/tecnologias utilizadas	5
2.1.2	Identificação das lacunas/melhorias a realizar	6
2.2	Sistemas TO-BE	6
2.2.1	Requisitos Funcionais	6
2.2.2	Requisitos técnicos e restrições ao desenvolvimento	6
2.3	Figuras	6
3	Desenvolvimento da aplicação	9
3.1	Análise e Especificações	9
3.1.1	Requisitos Funcionais	9
3.1.2	Requisitos Não Funcionais	10
3.2	Arquitetura técnica	10
3.2.1	Diagrama de Atividades	10
4	Análise dos Resultados	13
4.1	Sub-capítulo 1	13
4.2	Sub-capítulo 2	13
5	Conclusão Final	15
5.1	Sub-capítulo 1	15
5.2	Sub-capítulo 2	15

Lista de Figuras

1.1	Diagrama de Gantt	2
2.1	Esta é a legenda da figura	6
3.1	Diagrama de Atividades	11

Lista de Tabelas

3.1	Requisitos funcionais	10
3.2	Requisitos não funcionais	10

Listagens de Código

2.1	Função para substituir <code>mysql_result</code>	7
2.2	Função para substituir <code>mysql_field_name</code>	7
2.3	Improvement da função <code>nka_mysql_result</code>	7

Siglas & Acrónimos

MoSCoW Must Have, Should Have, Could Have, Won't Have. 9

PHP HyperText Preprocessor. 5

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

Glossário

MoSCoW Método de priorização de funcionalidades ou requisitos. 9

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.

– por terminar – acabaram as ideias !!

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, no entanto será necessário resolver alguns problemas de compatibilidade que poderão surgir consequentemente a esta alteração 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.

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

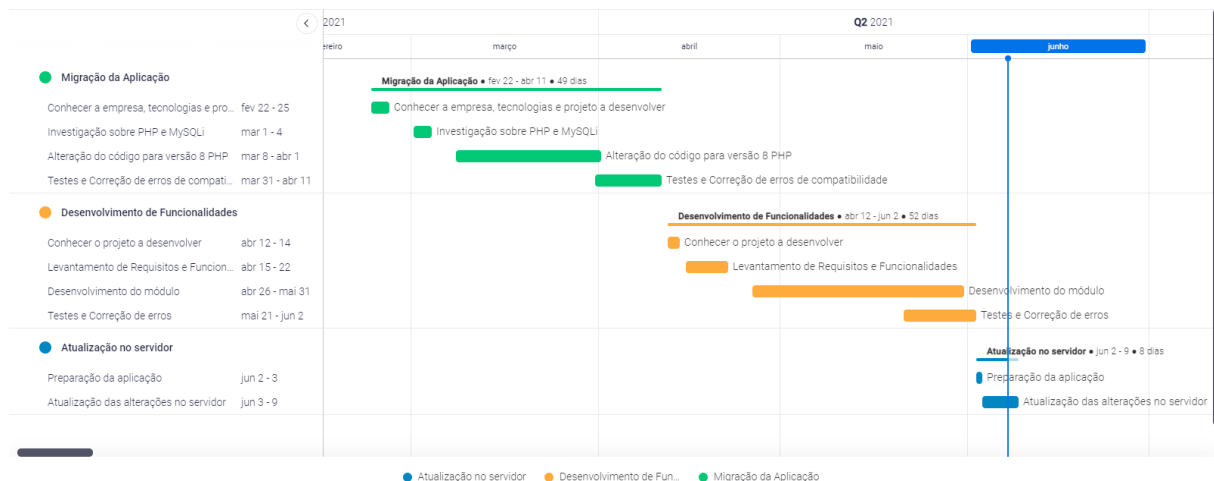


Figura 1.1: Diagrama de Gantt

1.4 Estrutura do documento

[A última secção da introdução deve explicar a estrutura do documento: quais são só capítulos existentes (para além do primeiro) e o que será discutido em cada um desses capítulos. A estrutura típica de um relatório de desenvolvimento de software é:

Introdução, com um breve resumo do que se pretende atingir, e uma descrição clara dos objetivos;

1. Análise ao problema, que poderá incluir uma análise ao estado da arte ou ao modelo de negócio onde se pretende intervir;

2. Análise e modelação do sistema, em que sejam levantados sistematicamente os requisitos, descritos diagramas de caso de uso e de atividade (que descrevam/formalizem o modelo de negócio).
3. Implementação, em que se descrevam as tecnologias escolhidas (e se justifiquem), e se refira detalhes sobre a implementação.
4. Análise de resultados e testes, seja uma análise/avaliação aos resultados obtidos, sejam testes de usabilidade ou unitários ao trabalho desenvolvido.
5. Conclusão.]

2. Processo de Reengenharia

2.1 Sistemas AS-IS

2.1.1 Descrição da Arquitetura/tecnologias utilizadas

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.

Xdebug

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

2.1.2 Identificação das lacunas/melhorias a realizar

2.2 Sistemas TO-BE

2.2.1 Requisitos Funcionais

2.2.2 Requisitos técnicos e restrições ao desenvolvimento

2.3 Figuras

Ao contrário do Word, o \LaTeX usa um mecanismo de colocação de figuras e tabelas em que estas flutuam ao longo das páginas de acordo com a necessidade/disponibilidade em termo de espaço vertical. Assim, não devem usar frases como “na figura acima”, ou “na figura abaixo”, mas fazer referências: “tal como se pode observar na Figura 2.1.”

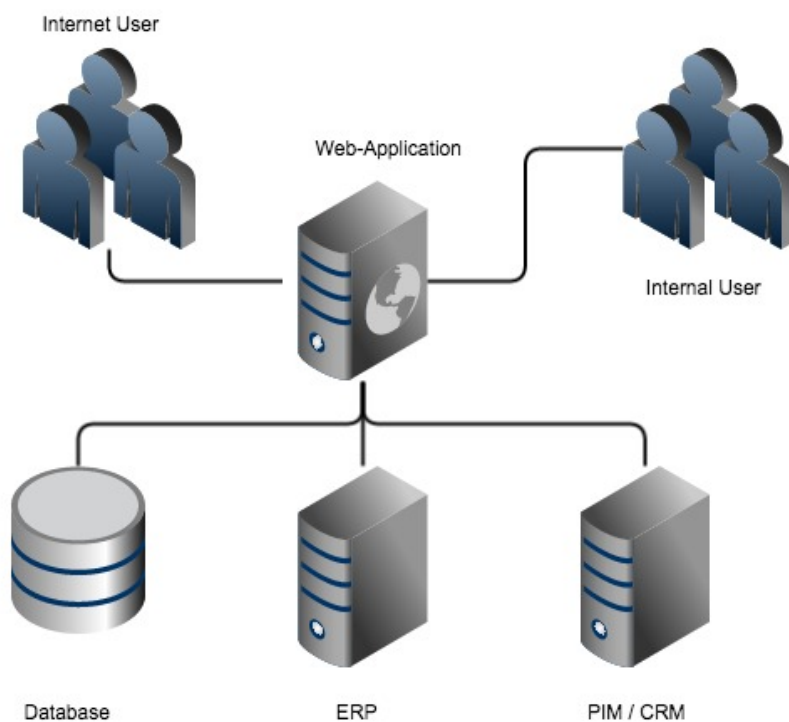


Figura 2.1: Esta é a legenda da figura

O mesmo acontece com as tabelas, como se pode ver na Tabela.

Para a inclusão de código, usa-se algo semelhante. Veja-se a Listagem 2.1.

```

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

```

Listagem 2.1: Função para substituir mysql_result.

```

if (!function_exists('mysql_field_name')) {
    function nka_mysql_result($res, $row, $field=0) {
        $res->data_seek($row);
        $datarow = $res->fetch_array();
        return $datarow[$field];
    }
}

```

Listagem 2.2: Função para substituir mysql_field_name.

```

if (!function_exists('nka_mysql_result')) {
    function nka_mysql_result($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 2.3: Improvement da função nka_mysql_result.

3. Desenvolvimento da aplicação

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:

MoSCoW

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

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

3.2.1 Diagrama de Atividades

Para descrever o processo de negócio foi utilizado diagramas de atividade, uma vez que permitem 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.

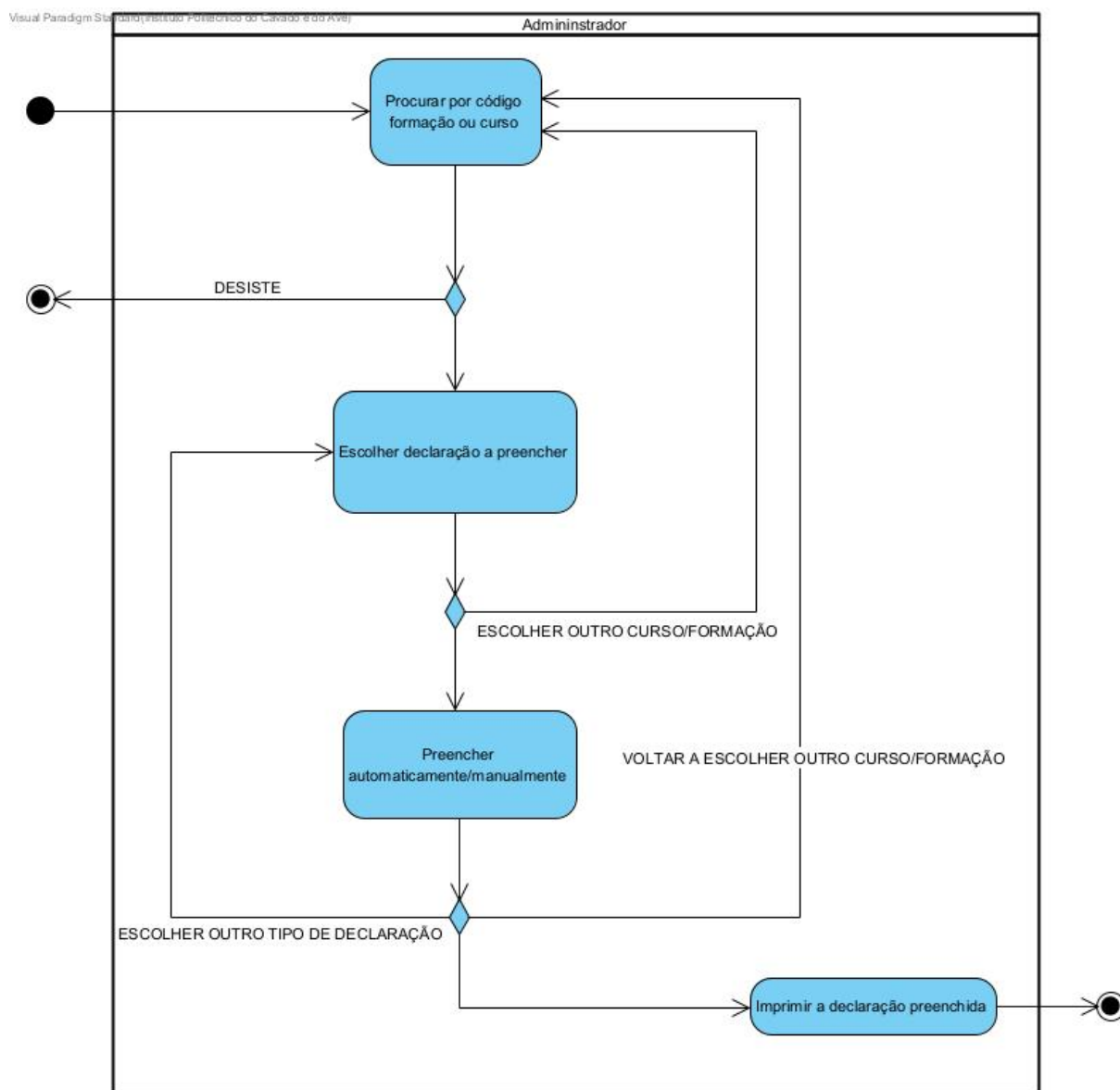


Figura 3.1: Diagrama de Atividades

4. Análise dos Resultados

4.1 Sub-capítulo 1

Texto do sub-capítulo 1

4.2 Sub-capítulo 2

Texto do sub-capítulo 2

5. Conclusão Final

5.1 Sub-capítulo 1

Texto do sub-capítulo 1

5.2 Sub-capítulo 2

Texto do sub-capítulo 2

Bibliografia