

Atividade Prática de Desenvolvimento Curricular - Estágio

Relatório de Estágio

Identificação

Ano Letivo	2020/2021	Semestre	1º	Estágio nº	4364
Aluno nº	52597				
Nome	Teresa Falcão de Campos Monteiro				
Título do estágio	SAP ABAP on the cloud				
Empresa	Accenture				
Coordenador	Paulo Ruivo				
Orientador do DI	Nuno Miguel Cavalheiro Marques				

Monitorização do Período Introdutório**Data: 06/02/2021****Local: Remotamente**

Objetivos e resultados esperados (3000 caracteres):

Resumo: O objetivo proposto foi o estudo de uma nova solução SAP-ABAP RESTFull programming model. Foram ilustrados os principais conceitos e foi estudada uma aplicação para uma agência de viagens. Os conceitos foram testados com a construção e análise comparativa de uma aplicação de teste.

O principal objetivo do estágio é perceber quais as vantagens e inconvenientes da adaptação da solução ABAP RESTfull programming model.

Para introduzir o estágio a Accenture deu duas formações.

O objetivo da primeira formação foi proporcionar uma melhor e mais fácil adaptação à linguagem ABAP e ao ambiente SAP ERP, de modo a preparar-me para a segunda formação. Para consolidar os conhecimentos foram disponibilizados alguns exercícios. Tive ainda a oportunidade de fazer esta formação numa das máquinas da Accenture, o que me permitiu ter uma maior noção do vocabulário e das funcionalidades do dia a dia da empresa.

No âmbito do objetivo principal do estágio, a Accenture tem como estratégia desenvolver soluções empresariais que não obrigue os clientes a mudanças estruturais, pelo contrário, sejam de fácil atualização nas aplicações. Essa solução pode ser o ABAP RESTfull programming model, que constou da segunda formação.

O modelo de programação ABAP RESTful, muito resumidamente, é o desenvolvimento de serviços RESTful OData que podem ser consumidos por aplicações SAP Fiori, ou por qualquer outro tipo de cliente Odata. Este oferece suporte ao desenvolvimento dos serviços OData otimizados para SAP HANA com base no core data services views. Este core data services views através das queries, e adicionando um behaviour definition para transactional basics, ficam a funcionar como as operações antigas: Create, Read, Update e Delete. O modelo de programação ABAP RESTful foi também integrado diretamente no core da linguagem ABAP, introduzindo novos tipos de objetos de desenvolvimento ABAP, transportáveis para behavior definitions, service definitions, e service bindings. Para além deste modelo de programação, também uma nova Entity Manipulation Language (EML) foi integrada no core da linguagem ABAP RESTful.

Esta estratégia poderá ser uma boa solução na medida em que podemos ter uma greenfield application ou uma brownfield application. Isto é, respetivamente, uma aplicação que pode ser construída do zero ou uma aplicação que consegue reutilizar pedaços de códigos de outras.

No final da segunda formação, no início de Dezembro (fim do período introdutório), fiz um teste de avaliação que me permitiu obter um certificado, que comprova que não só frequentei a formação, mas como também tive um aproveitamento superior a cinquenta por cento (no anexo 1 em baixo está o certificado). Durante a formação desenvolvi uma aplicação SAP Fiori App. A aplicação é sobre uma agência de viagens, ver no anexo 3 o layout da mesma.

Terminada a segunda formação aprofundei esta solução nomeadamente ao nível das mais valias para os projetos e para os clientes, identificado os respetivos pontos fortes ou pontos fracos. Para além de toda a documentação que me

foi disponibilizada durante a formação, também consigo perceber melhor, a um nível mais prático os benefícios para os projetos e para os clientes.

No final de Janeiro também fazia parte dos objetivos fazer uma apresentação para a equipa da Accenture para apresentar os tópicos referidos, como também descrever e explicar a forma como a aplicação SAP Fiori foi desenvolvida.

Descrição das atividades realizadas no período introdutório (3000 caracteres):

O primeiro dia de estágio iniciou-se com uma sessão de onboarding com todos os estagiários das universidades via teams, conduzida pelos recursos humanos da empresa. Nesta sessão foi nos feita uma breve apresentação sobre a empresa. Logo após esta sessão reuni com o meu orientador de estágio da empresa e comecei a ser guiada para a primeira formação, que teve a duração de duas semanas.

Nestas primeiras duas semanas estive a fazer uma formação sobre ABAP on cloud, que consistia em aprender esta nova linguagem de programação, que vou utilizar neste estágio. O foco desta formação foi aprender principalmente a sua sintaxe. Para isso tive uma lista de exercicios onde aprendia a:

- Criar programas;
- Criar tabelas;
- Inserir campos;
- Guardar;
- Criar dominios;
- Criar elementos de dados;
- Criar Programas Module Pool;
- Criação de Ajuda de Pesquisa;

Um dos outros objetivos de criar estas funcionalidades foi o de me familiarizar com o ambiente.

Relativamente à segunda formação (abap restfull programming model) teve a duração de 5 semanas. No final dessas semanas tive um exame que, por ter tido um resultado positivo me permitiu receber um certificado. Cada semana tive um conjunto de videos para ver, que estão divididos por unidades, e ao mesmo tempo desenvolver a aplicação (SAP Fiori).

Na primeira semana os conteúdos corresponderam à Introdução:

- Unit 1: The Big Picture
- Unit 2: Architecture Overview
- Unit 3: Involved Technologies
- Unit 4: Introducing SAP Cloud Platform, ABAP Environment
- Unit 5: Preparing Your ABAP Development Environment
- Unit 6: Creating Your First ABAP Cloud Console App

Na segunda semana o objetivo foi criar uma read-only list report application:

- Unit 1: The Business Scenario
- Unit 2: Creating the Database Tables
- Unit 3: Creating the Core Data Services (CDS) Data Model
- Unit 4: Defining the CDS Data Model Projection
- Unit 5: Enriching the Projected Data Model with UI Metadata
- Unit 6: Creating and Previewing the OData UI Service
- Unit 7: Implementing Basic Authorizations

É importante referir que existem dois tipos de aplicações. Nós temos a aplicação read-only que requer o core data services data model e uma aplicação analítica ou de pesquisa de anotações específicas. E temos ainda a Transactional applications que é um outro cenário, que adicionalmente ao que o read only necessita, requer a criação de um business object baseado num behavior definition e a implementação para manipular as operações CRUD (create, read, update and delete).

A terceira semana foi dedicada a Enabling the Transactional Behavior of an App:

- Unit 1: The Enhanced Business Scenario
- Unit 2: Defining the Basic Business Object Behavior
- Unit 3: Creating the Business Object Behavior Projection
- Unit 4: Understanding Entity Manipulation Language (EML)
- Unit 5: Enhancing the Business Object Behavior with App-Specific Logic

- Unit 6: Implementing the Business Object Behavior
- Unit 7: Enabling the Draft Handling
- Unit 8: Troubleshooting Your SAP Fiori App

Na quarta semana lidei com a fase do Dealing with Existing Code:

- Unit 1: The Business Scenario
- Unit 2: Creating the CDS Data Model
- Unit 3: Defining and Implementing the Business Object Behavior
- Unit 4: Creating the Business Object Projection
- Unit 5: Building and Previewing the OData UI Service

Finalmente, na quinta semana vamos tratar do Service Consumption e da Web APIs:

- Unit 1: The Business Scenario
- Unit 2: Creating the Service Consumption Model
- Unit 3: Defining the Custom CDS Entity
- Unit 4: Creating the Implementation Class
- Unit 5: Enhancing the Flight Model with External Data
- Unit 6: Creating and Previewing the OData UI Service
- Unit 7: Creating and Previewing the OData Web API
- Unit 8: Consuming the Web API with SAP Analytics Cloud

Dificuldades ou oportunidades encontradas, soluções adotadas, e sua justificação (6000 caracteres):

Neste período introdutório, relativamente à primeira formação a única dificuldade que poderia ter tido era a adaptação à linguagem, mas na prática até correu bem. Tive um vasto acesso à documentação e, para além disso, no início estive inserida numa equipa em que os colegas me esclareciam as dúvidas que surgiam. A outra dificuldade que tive foi a quantidade de informação que tinha a reter, muitos conceitos novos, e conseguir perceber como funciona a SAP ABAP on Cloud. Com a prática acabou por se tornar tudo muito mais intuitivo e fácil de interiorizar.

Ainda relativamente à primeira formação, como disse anteriormente na parte dos objetivos e resultados esperados, a máquina que eu estava a utilizar expirou, não tendo tido oportunidade de guardar os resultados do trabalho feito. Podia ter recommçado, mas como ia iniciar a segunda formação não se justificava repetir a aprendizagem em sintaxe.

Na segunda formação, já tive mais dificuldades a perceber como é que o ambiente funcionava. Usar o eclipse para mim não foi um entrave, porque é uma ferramenta com que estou muito habituada a trabalhar, mas associá-lo a uma aplicação já foi mais desafiante. Nesta fase, também tinha bastante documentação o que foi benéfico e fácil para avançar.

Aproveito para salientar um aspeto relacionado com o ambiente de trabalho e com a pandemia que atravessamos. Inicialmente eu estava inserida numa equipa, porque estava a receber a mesma formação que outros estagiários. No entanto, quando essa formação acabou e comecei uma nova, comecei a trabalhar mais sozinha. Apesar de ter sempre alguém para me ajudar e estar sempre acompanhada, senti falta do ambiente de equipa.

Plano de trabalhos (3000 caracteres):

O estágio começou no dia 19 de Outubro de 2020 e foram definidos os seguintes pontos:

1. Formação inicial SAP Abap on Cloud: (primeiras duas semanas)
 - Criar programas;
 - Criar tabelas;
 - Inserir campos;
 - Guardar;
 - Criar domínios;
 - Criar elementos de dados;
 - Criar Programas Module Pool;
 - Criação de Ajuda de Pesquisa;
2. Posteriormente surgiu outra formação sobre abap restfull programming model; (duração de cinco semanas)
 - 2.1. Lista das funcionalidades que permite a construção da aplicação:
 - Core data services;
 - Behaviour definition;
 - Business objects;

- Service definition;
- Service binding;
- Dictionary;
- Text messages;
- Value help

2.2. Resolução do teste para obtenção do certificado; (um dia)

3. Explorar a linguagem e a solução para perceber as vantagens;
4. Apresentação e demonstração, para a equipa, do trabalho desenvolvido durante o estágio. Esta apresentação, tem como principal objetivo explicar, detalhadamente, a aplicação desenvolvida, e mostrar que vantagens e desvantagens há em investir nesta solução.
5. Relatório final e apresentação final.

A este ponto do relatório, acabei o segundo ponto do plano.

Formação recebida e conhecimentos adquiridos (máx. 3000 caracteres):

Como já disse anteriormente, a Accenture disponibilizou-me duas formações.

A primeira formação já estava a decorrer com outros dois colegas quando cheguei, no entanto entrei para uma equipa e comecei. Aprendi a linguagem abap, que era para mim uma linguagem nova. Esta formação permitiu-me solidificar bem os novos conceitos, a forma como escrever e inserir-me no SAP ERP (a máquina). Apesar da linguagem ser diferente, não achei a formação muito difícil porque os exercícios foram muito à base de raciocínio, o que é equivalente ao que se passa e passou em muitas disciplinas do meu curso. Tive a oportunidade de trabalhar numa das máquinas da empresa, o qual me ajudou a perceber melhor a teoria.

Relativamente à segunda formação aprendi ainda mais conceitos novos, o que nem sempre foi fácil, porque significa muita informação nova em pouco tempo. Porém esta formação estava muito bem organizada, porque explica passo a passo, nomeadamente, tudo o que se deve fazer para criar uma aplicação SAP Fiori (o seu development flow). No anexo 2 em baixo, está o esquema do mesmo.

Para além disso, explica o que significa cada funcionalidade e qual a função dela. As definições de todos estes conceitos estão apresentados no anexo – aplicação de agência de viagens na parte do glossário.

Adicionalmente, na formação também se fez uma análise geral do como a SAP tem evoluído, o que me ajuda muito a perceber qual será o benefício ou não de adotar esta estratégia. A acrescentar os conhecimentos adquiridos, explica como é que é composta em termos de arquitetura e como é que as componentes referidas anteriormente se interligam. Mostra também quais são os principais focos da aplicação. O foco principal é a experiência dos utilizadores e se estes de facto gostam de usar a aplicação. Para isso a aplicação deve ser simples, pois queremos obter o resultados que o utilizador pretende com o mínimo de “clicks” possíveis. É preciso também ser coerente em todos os dispositivos, adaptável para não desformatar o aspeto, delightful, para retirar o que não é relevante para o trabalho, impendendo distrações e enganos, e sem descurar o role based, que consiste em que a aplicação ofereça apenas o que é relevante para o utilizador.

Descrição técnica preliminar (máx. 4000 caracteres)

Nesta fase, surgiu um imprevisto, que será explicado nos desvios do plano inicial e por isso tive que recomeçar de novo a aplicação, cuja origem é um serviço contratado à Accenture por uma Agência de viagens. Acabei por usar o tempo desta fase para a desenvolver de novo. Vou passar a explicar a construção da mesma, cujo o esquema desta explicação é possível ver no anexo 2 (esquema do development flow).

O primeiro passo é o modelo de dados com a database armazenada no dicionário ABAP que é criada pela database da SAP HANA.

Depois temos o core data service que independentemente se a aplicação é do tipo read only ou transacional, é a próxima etapa. As core data services views dão nos uma visão de como são as estruturas da tabela de base de dados e as suas relações.

De seguida, transforma-se um modelo de dados core data services num business object. O modelo core data services precisa de definir uma hierarquia. Com o modelo de programação ABAP RESTful, a definição da hierarquia foi totalmente integrada no core da linguagem ABAP (substituindo o uso de anotações no Business Object Processing Framework (BOPF)). Para adicionar o comportamento real do business object (operações create, read, update e delete e ações personalizadas), é necessário um behavior definition com um behavior implementation correspondente.

O behavior definition passa a ser o novo objecto que vai definir quais as ações de cada business object. A implementação dessas operações é controlada pelo behavior implementation.

O nosso modelo de dados apenas se encontra na ABAP stack, que para ser usado pela SAP Fiori application ou outro Odata client, tem que ser exposto ao Odata service.

Com o ABAP RESTful programming model, a definição do core data service views e as entidades OData foram integradas na linguagem ABAP na forma de uma service definition. Dentro da service definition podemos de maneira declarativa através de uma sintaxe semelhante ao do core data service, definir quais, entidades a expor com o nosso serviço Odata, sem a necessidade de transações ou ferramentas SAP adicionais.

Com o service binding podemos expor um modelo da aplicação ligando o service binding a um protocolo de comunicação cliente-servidor específico, como é o caso do Odata. Este tipo de ligação do service binding oferece atualmente o suporte a dois cenários: um serviço de user interface (UI) ou uma API da web. Um serviço de UI é um serviço OData que permite adicionar ou uma UI de elementos SAP Fiori, ou outros clientes UI ao serviço, ao mesmo tempo que incorpora anotações da UI, enquanto uma API da web é um serviço OData puro usado para extensões imprevistas do software SAP.

Posteriormente, vamos criar o business object definition. E definir o standard transactional Business object behavior para cada entity. Depois vamos criar um business object behavior projection. O behavior projection é quem projeta os recursos transacionais a partir da behavior definition básica. Teremos também o behavior definition projection que projeta as operações: o create, update e o delete.

Na parte do abap code são mostradas mensagens de erros quando as informações inseridas não correspondem ao que é suposto. (ver anexo 7.1)

Outra funcionalidade a implementar é uma funcionalidade que na minha opinião é bastante importante porque não nos obriga a estarmos sempre a gravar na database todos os pequenos movimentos. A funcionalidade draft equivale por exemplo ao papel de rascunho do email: quando estamos a escrever um email e ainda não o queremos enviar, ele fica guardado até o utilizador ir ao rascunho e decidir usá-lo ou não. Aqui acontece o mesmo, o draft permite-nos começar a fazer uma reserva, mas só guardamos na database quando queremos tomar uma decisão decisiva.

A última funcionalidade a ser implementada é o value help que consiste numa ajuda no preenchimento dos dados de uma reserva. Este funciona da seguinte maneira: primeiro fazemos download dos metadados dos ficheiros que contém o modelo de dados da entidade do serviço remoto OData. Em seguida criamos o service consumption model. Depois, vamos criar uma entidade core data service personalizada, aproveitando as entidades abstratas que estão a ser geradas no service consumption model que criamos.

Finalmente, criamos uma entidade personalizada, onde teremos que implementar a classe ABAP que tem a implementação de nossa query. É aqui que é realizada a chamada para o serviço OData remoto.

Depois de ter criado a entidade personalizada, podemos aprimorar o modelo de dados da nossa managed application, de modo a que a entidade personalizada seja usada como auxiliar no caso de não sabermos algum dado. Usamos também a validação de dados do cliente, que também chamará o serviço OData remoto. No anexo 8 podemos ver o que é o value help e no 8.1 o esquema do mesmo.

Atualização do plano de trabalhos (máx. 3000 caracteres):

Como referi na descrição técnica preliminar acima, tive que recomeçar de novo a aplicação que me ocupou durante as três primeiras semanas de Janeiro. O plano dos trabalhos foi:

1. Lista das funcionalidades que permitem a construção da aplicação: (primeira, e metade da segunda semana)
 - Core data services;
 - Behaviour definition;
 - Business objects;
 - Service definition;
 - Service binding;
 - Dictionary;
 - Text messages;
 - Value help
2. Explorar a linguagem e esta solução para perceber as vantagens; (metade da segunda semana e terceira semana)
3. Apresentação e demonstração, para a equipa, do trabalho desenvolvido durante o estágio (1 dia). Esta apresentação tem como principal objetivo explicar detalhadamente, a aplicação desenvolvida e mostrar quais as vantagens e os inconvenientes que existem ao investir nesta solução.
4. Relatório final e apresentação final.

Desvios ao plano inicial, dificuldades endereçadas e sua justificação (máx. 3000 caracteres):

No estágio utilizei o SAP Cloud Platform, ABAP environment system e como é para exercicios de um curso, o sistema que uso é o trial systems. Este tem um limite de dias, isto é, ao fim de 30 dias perco todo o trabalho. Antes do periodo de interrupção do Natal e Ano Novo eu já tinha acabado a aplicação que verifiquei estar a executar bem, sem erros e com a aplicação SAP Fiori a funcionar na sua totalidade. Porém quando retomei o estágio todo o meu trabalho tinha sido apagado, pelo que a solução que encontrei foi desenvolver a aplicação toda de novo. Ao desenvolver de novo, quando estava a fazer o que correspondia à quarta semana da formação a aplicação não funcionou como devia. Nesta semana tinha que criar um novo packet com todas as funcionalidades já feitas noutro packet mais as novas, porque no desenvolvimento da aplicação tanto podemos ter uma aplicação do tipo managed implementation, como unmanaged implementation. Então, esta quarta semana foi para desenvolver a aplicação do tipo unmanaged implementation. O problema surgiu quando fui correr a aplicação no browser e esta não funcionou na sua totalidade, apenas apareciam metade das palavras. Não consegui perceber porque é que os resultados não apareciam conforme se pretendia relativamente a esta parte, pois já tudo tinha funcionado bem da primeira vez. O código estava sem erros, não tinha warnings e todas as ligações foram feitas corretamente, isto é, segui os mesmos passos da mesma forma que anteriormente. A alternativa que arranjei para este impasse foi perceber bem o que estava a fazer e adaptar a um primeiro packet que estava a funcionar como é esperado.

Apesar desta imprevisto o plano inicial não foi afetado, pois tentei trabalhar em paralelo nas tarefas de reconstrução da aplicação e de recolha de informações para o objetivo final.

Descrição final do trabalho efetuado:**Objetivos (máx. 3000 caracteres)**

Relativamente aos objetivos definidos na fase introdutória e intermédia do estágio, tudo permaneceu igual e todos conseguiram ser cumpridos.

O objetivo principal do estágio era perceber se a solução de Abap RESTfull programming model é uma vantagem a adotar, e se vale a pena aplicá-la em projetos e promover junto dos clientes.

Mais precisamente, relativamente à apresentação tinha que tocar nos seguintes tópicos:

- Fazer uma introdução sobre o que é ABAP RESTfull programming model;
- Falar sobre como é constituída e qual a sua arquitetura;
- Falar sobre a sua evolução, onde indiretamente refiro algumas vantagens e desvantagens;
- Explicar como a aplicação é desenvolvida (development flow);
- Apresentar a aplicação SAP Fiori desenvolvida.

Todas as funcionalidades que referi neste relatório são desenvolvidas ao longo da aplicação SAP Fiori. São referidas também na descrição da arquitetura e no development flow.

Descrição técnica e seu contexto (máx. 10000 caracteres)

Como estagiária não tive a oportunidade de trabalhar com o cliente, por isso foi-me dado uma espécie de case study. O objetivo era construir uma aplicação para uma agência de viagens. Como disse anteriormente na fase introdutória, o foco está na experiência e nas necessidades da agência de viagens. A aplicação deve permitir que o utilizador possa pesquisar e filtrar informação na sua base de dados, por exemplo, através do agencyID ou do travelID ou do CustomerID (anexo). Para além disso, deve permitir também criar uma nova viagem e dentro de cada viagem poder também criar novas reservas associadas, ver anexo 6 e 7 para ver as pesquisas e para criar. Deve deixar apagar tanto as viagens como as reservas, assim como permitir ver os detalhes das reservas e de viagens específicas, ver anexo 4 e 5 respetivamente.

Além disso, deve ter um sistema de value help onde conseguimos encontrar a informação (dados) em falta, e poder mudar também o layout onde vemos a informação do cliente, como por exemplo:

- mostrar o preço total, isto é o conjunto da viagem mais os bookings associados;
- qual a moeda associada;
- qual o UUID;
- quando é o início e o fim da viagem;
- a descrição associada;
- qual a fee do booking;
- ver anexo 9 para mais exemplos;

Deve permitir ainda poder aceder aos drafts do operador, isto é rascunhos, como este acesso a ser permitido através de um dos filtros (all status) (ver anexo).

Outras funcionalidades são aceitar ou rejeitar a viagem, e indicar erros quando estamos a introduzir os dados do cliente na base dados de forma incorreta.

Como já expliquei na fase intermédia quais os passos a seguir para a construção da aplicação (development flow), penso que o que faz mais sentido nesta descrição técnica são as interações. A nível de código criei no total três packets. O primeiro packet, que corresponde às primeiras três semanas da segunda formação que refiro no plano de trabalhos, insere-se numa aplicação do tipo managed implementation. O segundo packet desenvolvido na quarta semana da formação, corresponde a um tipo de implementação unmanaged. Ambas as implementações têm um resultado final parecido a nível de aspeto. O terceiro packet está relacionado com uma possível adição de metadados. Este terceiro packet é o que permite ter a funcionalidade value help.

Mais detalhes já foram referidos na descrição técnica preliminar. Nesta descrição são explicados todos os passos tanto do primeiro packet como do segundo. Apesar de serem dois tipos de implementação diferentes, a estrutura de construção é a mesma. Já o terceiro packet, que é a implementação do value help, é explicado nos últimos três parágrafos da descrição preliminar. Este tem uma estrutura de construção diferente, ver anexo.

Uma vez que no desenvolvimento da aplicação de agência de viagens estava a ser acompanhada pela formação, o que escrevi na descrição preliminar não desviou praticamente em nada do que era o resultado final, pelo que não vale a pena ser novamente detalhado na descrição técnica.

Estado de desenvolvimento final das tarefas do Plano de Trabalhos (máx. 3000 caracteres):

Todos os objetivos foram cumpridos e até com sucesso. Nesta fase já fiz a apresentação para a empresa Accenture e obtive um bom feedback.

A duração das tarefas, durante este estágio, foram as seguintes

1. Formação inicial SAP Abap on Cloud (primeiras duas semanas)
 - Criar programas;
 - Criar tabelas;
 - Inserir campos;
 - Guardar;
 - Criar domínios;
 - Criar elementos de dados;
 - Criar Programas Module Pool;
 - Criação de Ajuda de Pesquisa;
2. Segunda formação, abap restfull programming model (duração de cinco semanas)
 - 2.1. Lista das funcionalidades que permite a construção da aplicação
 - Core data services;
 - Behaviour definition;
 - Business objects;
 - Service definition;
 - Service binding;
 - Dictionary;
 - Text messages;
 - Value help.
6. Resolução do teste para obtenção do certificado (1 dia)
7. Refazer a aplicação do início e por isso a repetição das mesmas funcionalidades acima referidas (2 semanas)
8. Explorar a linguagem e esta solução para perceber os benefícios; (3 semanas, em conjunto com o ponto 9)
9. Apresentação e demonstração, para a equipa, do trabalho desenvolvido durante o estágio. Esta apresentação, tem como principal objetivo explicar, detalhadamente, a aplicação desenvolvida, e mostrar que vantagens e desvantagens há em investir nesta solução; (1 dia)

Resultados obtidos (máx. 3000 caracteres)

O primeiro resultado principal obtido foi a aplicação SAP Fiori de agência de viagens que desenvolvi.

Os artefactos que foram concebidos na aplicação foram

- Core data services;
- Behaviour definition;
- Business objects;
- Service definition;
- Service binding;
- Dictionary;
- Text messages;
- Operações de create, update;
- Value help.

Após uma extensiva pesquisa sobre vantagens e inconvenientes da ABAP RESTfull programming model conclui alguns aspetos positivos. O Abap RESTfull programming model (RAP) lida com API externa com conteúdo core data services. E um dos recursos principais do RAP é a base de dados HANA fazendo com que a performance seja superior, o que é um ótimo benefício e algo muito procurado.

Antes de referir os aspetos negativos acho importante definir o que é uma implementação managed. Uma implementation managed está associada a uma greenfield application, definida anteriormente, que tem o runtime managed. Devido a este tipo de implementação existem algumas restrições como por exemplo a nível numérico. A numeração é atrasada e por isso não é compatível com o tipo de implementação. Também algumas ações estão restringidas, como por exemplo a primary key que não pode usar valores com mais de cento e dezassete caracteres ou