

Sheet1

1	<p>grupoav.: 1 sol.....: Suficiente uml.....: Bom coment...: O UML está bom, mas o relatório está um pouco sucinto de mais. Talvez ficasse melhor se comentassem as figuras, embora estas sejam esclarecedoras.</p>	<p>grupoav.: 1 sol.....: Suficiente uml.....: Suficiente coment...: Penso que deviam ter explicado melhor a solução. Apenas apresentaram os diagramas. Os diagramas de estados não reflectem todas as operações do sistema</p>
2	<p>grupoav.: 2 sol.....: Bom uml.....: Muito Bom coment...: ## Avaliação da aplicação</p> <p>A aplicação apresenta todas as funcionalidades exigíveis a este tipo de aplicação informática. Embora como ja&#8217; referi no comentário anterior não esta&#8217; especificado o backup da informação gerida pela aplicação.</p> <p>## Avaliação dos use cases</p> <p>O UML apresentado esta&#8217; muito bom, e bem organizado/documentado. Achei interessante a maneira iterativa apresentada neste relatório ate&#8217; chegar a solução final.</p>	<p>grupoav.: 2 sol.....: Bom uml.....: Muito Bom coment...: O relatório está bom, mas acho que não era necessario apresentar tantos diagramas de classe,podia estar mais sucinto.</p>
3		
4		

Sheet1

<p>grupoav.: 1 sol.....: Suficiente uml.....: Suficiente coment...: -Falta uma descrição textual do diagrama de classes para ajudar a entender. -Sistema nao suporta fragmentação,nem distinção entre tarefa móvel ou fixa.</p>	<p>grupoav.: 1 sol.....: Suficiente uml.....: Bom coment...: A qualidade do UML apresentado é boa mas a qualidade da solução poderia ser melhor, uma vez que considerado que está demasiado simples. Acho que tendo em conta a solução da fase 1 haveriam mais algumas classes que poderiam ser criadas e também acho que poderiam ter sido criados mais alguns métodos que dessem mais opções ao utilizador. De resto está bastante perceptível e bem estruturado o que ajuda a perceber o que pretendem implementar e o relatório também está muito bem organizado.</p>	
<p>grupoav.: 2 sol.....: Bom uml.....: Bom coment...:</p>	<p>grupoav.: 2 sol.....: Suficiente uml.....: Muito Bom coment...:</p>	

Sheet1

<p>5</p>	<p>grupoav.: 5 sol.....: Fraco uml.....: Fraco coment...: 7</p> <p>A classe Sistema diz que implementa a interface Utilizador e não implementa todos os métodos dessa mesma interface. Dizem também que o sistema é composto por todas as outras tarefas, quais?</p> <p>São usadas setas de associação por exemplo entre as classa Tarefa e Alarme, que representam bidireccionalidade entre as classes, mas acho que não faz sentido em Java.</p> <p>Se a classe Data não guarda uma lista de tarefas, como é que consegue devolver um lista de tarefas com o metodo getTarefas?</p> <p>É descrito e muito bem que o UML permite a gereção de código automaticamente, mas com a solução apresentada, contendo duas classes iguais mas que não podem ser uma só, cria um conflito em Java, na geração de código.</p> <p>Na discussão das vantagens/desvantagens da solução desenvolvida, falam de classes que não existem na solução.</p>	<p>grupoav.: 5 sol.....: Suficiente uml.....: Bom coment...:</p>
<p>6</p>	<p>grupoav.: 6 sol.....: Bom uml.....: Bom coment...: Penso que nesta fase o grupo melhorou evidentemente a proposta. No entanto gostaria de apenas de questionar as seguintes opções: -não vejo grande utilidade na existência das classes Data e Hora, uma vez que existe já em java classes que suportam estes dois "tipos" conjuntamente. Não obstante que nesta fase ainda seja compreensível esta opção. -para mim também não faz muito sentido a classe Tipo, pois como é dito que serve para ter uma só String como estrutura a indicar tratar-se de uma reunião, aula, ou outra coisa do tipo, poderia esta informação claramente ser incluída na classe Tarefa.</p>	<p>grupoav.: 6 sol.....: Bom uml.....: Muito Bom coment...:</p>

Sheet1

<p>grupoav...: 5 sol.....: Bom uml.....: Bom coment...: Em comparação com a fase anterior, o projecto parece muito melhor. A descrição da solução encontrada é muito mais esclarecedora. Apenas acho que deveriam tentar resolver o problema de ter a classe &#8220;Tarefa Dependente&#8221; repetida 2 vezes. Não me parece muito eficiente ter a mesma coisa repetida 2 vezes. O projecto responde claramente aos requisitos básicos que foram apresentados.</p>		
<p>grupoav...: 6 sol.....: Bom uml.....: Bom coment...: As classes, as relações entre as mesmas e os métodos a estas associadas parecem-me bem definidos. Penso que os objectivos pretendidos para esta fase foram atingidos.</p>	<p>grupoav...: 6 sol.....: Bom uml.....: Bom coment...: gostei do diagrama de classes apesar de achar que os métodos da classes, não eram necessários nesta fase pois não foram feitos os diagramas de interacção.</p>	<p>grupoav...: 6 sol.....: Muito Bom uml.....: Muito Bom coment...: Relativamente à solução, creio que focam todos os aspectos importantes, e está essa solução bastante bem definida. No que diz respeito à qualidade do UML, axo que é óptima, assim como o relatório está bastante elucidativo em relação ao que o grupo tem em mente para o trabalho</p>

Sheet1

7	<p>grupoav.: 7 sol.....: Bom uml.....: Bom coment...: Nos atributos têm a Duração como um dos atributos o que nao tem logica, visto existir as horas e datas iniciais e finais, logo nao tem logica, visto ate nao falarem mais neste atributo. o UML parece estar bom. Abraço []</p>	<p>grupoav.: 7 sol.....: Bom uml.....: Suficiente coment...: Penso que de acordo com o proposto na fase 1 do trabalho, foi apresentada uma solução capaz de implementar todas as funcionalidades. Apenas tenho a comentar que no relatorio não é explicada qual a função da classe aviso, e como esta não se encontra "ligada" à classe tarefa (como seria esperado) não se percebe muito bem a sua funcionalidade. Ainda no UML penso que a ligação entre Tipo -> Tarefa deveria ser no sentido inverso.</p>
8	<p>grupoav.: 8 sol.....: Fraco uml.....: Muito Fraco coment...: O Diagrama de classes está bastante incompleto, não apresentando uma solução total para o problemas, eles apenas mostram o objecto Tarefa e mais nada adiantam sobre o facto de como implementam o projecto.</p>	
9	<p>grupoav.: 9 sol.....: Suficiente uml.....: Bom coment...: Podiam ter feito uma melhor descrição das classes que utilizam, bem como uma referencia às suas utilidades.</p>	<p>grupoav.: 9 sol.....: Bom uml.....: Bom coment...: Penso que devia ter uma ligacao da classe tarefa para ela propria para indentificar dependencias... mas como na primeira fase penso que tambem nao contava com as dependencias, ajusta-se aos use cases da primeira fase. Penso que o UML esta bom e se ajusta a primeira fase, gostei da adicao dos diagramas de estado, embora pense que ate foram demasiados para a fase em que estamos. A inclusao da maioria/totalidade dos metodos de classe nesta fase acho que so complica a visualizacao (diagramas GRANDES). Nota: por favor nao entregem em Word, torna-se dificil visualizar direito... openoffice.....d</p>
10	<p>grupoav.: 10 sol.....: Fraco uml.....: Suficiente coment...: Relatorio sucinto mas claro. As variaveis data e hora, na classe Tempo deveriam ser do tipo Date ou Int, de modo a facilitar a manipulação dos valores. Boa ideia de criar a classe Info_Crono. Relação entre registo_D e List_T e Tarefa errada.</p>	<p>grupoav.: 10 sol.....: Fraco uml.....: Fraco coment...: No uml falta multiplicidades,agregações e tem classes a mais como no caso da classe lista_t de tarefas. As vantagens e desvantagens mais uma vez não se enquadram neste contexto.Tinha mais sentido se pusessem as vantagens do vosso uml e da vossa solução.</p>

Sheet1

<p>grupoav...: 7 sol.....: Bom uml.....: Bom coment....:</p>	<p>grupoav...: 7 sol.....: Bom uml.....: Bom coment...: Esta fase encontra-se dentro do pre-estabelecido, simples e objectiva, estando bem definidos os objectivos a implementar. Não usa multiplicidade nem tem nome nas associações. O UML encontra-se simples. A solução cumpre os requisitos.</p>	
<p>grupoav...: 9 sol.....: Bom uml.....: Bom coment...: O relatório está suficiente visto que não clarifica o uso das classes e métodos.</p>	<p>grupoav...: 9 sol.....: Bom uml.....: Bom coment...:</p>	
<p>grupoav...: 10 sol.....: Muito Bom uml.....: Muito Bom coment...: Nesta etapa o grupo soube implementar bem os diagramas de classe, exprimindo claramente a sua ideia em relação ao sistema de gestão por si a desenvolver. Perante o relatório bem estruturado e com uma solução bem explicada, atribui a nota de MUITO BOM para a solução encontrada e MUITO BOM para a qualidade de UML apresentada.</p>	<p>grupoav...: 10 sol.....: Suficiente uml.....: Bom coment...:</p>	

Sheet1

11	<p>grupoav.: 11 sol.....: Bom uml.....: Bom coment....:</p>	<p>grupoav.: 11 sol.....: Bom uml.....: Bom coment....: O relatório está um pouco curto, mas a solução parece-me bem.</p>
12	<p>grupoav.: 12 sol.....: Bom uml.....: Bom coment....: UML bastante detalhado e com bons pormenores. A divisao dos diagramas foi oportuna e bem conseguida.</p>	<p>grupoav.: 12 sol.....: Muito Bom uml.....: Muito Bom coment....: A qualidade da solução apresentada está francamente boa e funcional por isso a nota não poderia ser outra. A qualidade do UML também está muito boa, sem esquecimentos e muito completa.</p>
13	<p>grupoav.: 13 sol.....: Bom uml.....: Suficiente coment....: Penso que deveria ter diagramas de estado dos metodos que modificam a tarefa, para mais facil compreensão da solução. Tem uma introdução que engloba o essencial da fase em questão, o que é bastante positivo.</p>	<p>grupoav.: 13 sol.....: Bom uml.....: Bom coment....: Solução: nao percebi a utilidade das chamadas classes principais visto que ja existem outras classes que permitem a interface com o utilizador UML: parece estar de acordo com a solução proposta</p>
14	<p>grupoav.: 14 sol.....: Suficiente uml.....: Bom coment....:</p>	<p>grupoav.: 14 sol.....: Muito Bom uml.....: Muito Bom coment....:</p>

Sheet1

<p>grupoav.: 11 sol.....: Muito Bom uml.....: Bom coment...: - Não me parece muito pertinente de considerar Lembrete como superclasse das classes Aviso e Tarefa. - De facto se a classe Lembrete é superclasse das classes Aviso e Tarefa, então a seta de agregação entre a classe Agenda e classe Aviso devia estar orientada para a classe Lembrete. - Fazia mais sentido se a classe Tarefa fosse agregada a classe Agenda. A agenda contém tarefas não.</p>		
<p>grupoav.: 12 sol.....: Muito Bom uml.....: Muito Bom coment...: A solução esta bastante bem apresentada. Penso que apenas seja de referir que era favorecedor terem incluído no relatório um esquema das classes pelo menos para o sistema em geral(uma vez que fazer isso para cada use case seria efectuar sucessivas repetições na maioria dos casos).</p>	<p>grupoav.: 12 sol.....: Suficiente uml.....: Suficiente coment...: Mais uma vez nota-se um relatório pouco organizado sem introduções, conclusões e sem o próprio diagrama de classes. Também falta a descrição de alguns métodos (realocar tarefa). Relativamente à solução parece-me que há classes a mais que poderiam ser evitadas, não deixa de ser uma solução razoável mas um bocado confusa.</p>	
<p>grupoav.: 13 sol.....: Bom uml.....: Bom coment...: Mais cuidado com o relatório possibilitou uma melhor facilidade de compreensão da proposta. A solução é boa, acho que estão a utilizar poucas classes o que vai tornar o projecto final mais confuso, tarefa fixa e tarefa móvel não são muito esclarecidas, se a tarefa não é uma classe abstracta. Creio que as classes GEstaoTarefas e GestaoAviso vão precisar de mais algumas coisas! Mas, como já disse, acho que a solução esta muito bem apresentada.</p>		
<p>grupoav.: 14 sol.....: Bom uml.....: Bom coment...: A única coisa que poderia ter sido incluída seria uma breve descrição do UML e as vantagens e desvantagens da sua utilização nesta fase bem como as vantagens e desvantagens do modelo apresentado. O que resta penso que está bem explícito e com uma linguagem acessível a qualquer potencial cliente do gestor de tarefas.</p>		

Sheet1

<p>15</p>	<p>grupoav.: 15 sol.....: Bom uml.....: Bom coment...: Solução bem elaborada. UML concordante com a solução(use cases) apresentada no primeiro trabalho. Boa solução.</p>	<p>grupoav.: 15 sol.....: Muito Bom uml.....: Muito Bom coment...: O relatorio esta muito bom, sendo explicadas ao pormenor todas as classes criadas, bem como os metodos e construtores de cada uma. Penso que as interfaces criadas estao apropriadas relativamente a solução proposta. Conseguiram transmitir bem todas as ideias.</p>
<p>16</p>	<p>grupoav.: 16 sol.....: Bom uml.....: Suficiente coment...: Aspectos Positivos: -Definicao elaborada dos métodos e variáveis usadas -Classes bem estruturadas no diagrama, dividindo as tarefas em duas sub-classes: Tarefas escalonadas e Tarefas fixas Aspectos Negativos: - Má definição de interface. É referido que a interface Utilizador é abstracta pois os seus métodos nunca são directamente implementados. No entanto nunca se define directamente os métodos na respectiva interface mas sim nas classes que a implementam. É também dito que as classes que implementam esta interface são sub-interfaces o que não é verdade.</p>	<p>grupoav.: 16 sol.....: Bom uml.....: Bom coment...: Comentários negativos: 1. A introdução não introduz verdadeiramente ao trabalho realizado. Remete a contextualização deste trabalho a documentos externos o que deixa o leitor sem uma noção do que foi desenvolvido. 2. Ainda não é explícito o que é uma tarefa escalonada. Mais uma vez se sub-entende que é uma tarefa que é agendada automaticamente pelo sistema. 3. Não é claro como são introduzidas as tarefas no mapa da agenda. A chave é o nome, data, prioridade? 4. Não faz referência à classe que vai agendar automaticamente a tarefa. Segundo o relatório é a classe Escalonada que contém todos os métodos para escalonar uma tarefa, mas não é apresentado um método 'escalonar' ou 'agendar' Comentários positivos: 1. O relatório está sucinto e cobre todos os aspectos pretendidos nesta fase. 2. O modelo é simples e extensível o que é que se revela uma boa política cada vez mais visível à medida que se avança no trabalho. 3. Os diagramas são claros e fáceis de consultar.</p>

Sheet1

grupoav.: 15 sol.....: Suficiente uml.....: Bom coment....:		

Sheet1

<p>17</p>	<p>grupoav.: 17 sol.....: Suficiente uml.....: Suficiente coment....:</p>	<p>grupoav.: 17 sol.....: Fraco uml.....: Suficiente coment....: Achei original a ideia de existir o conceito de esquema de horário, associável a cada dia, mas falta explicação sobre a sua aplicação. Por exemplo, é possível alocar tarefas fora desse horário? quais são as restrições que esse horário implica? É necessário ter um horário associado a um dia? (dúvidas em parte herdadas da fase 1)</p> <p>Poderia ter mais documentação, o relatório é minimal.</p>
<p>18</p>	<p>grupoav.: 18 sol.....: Suficiente uml.....: Fraco coment....: Considero a solução suficiente visto terem apenas uma abordagem relacional entre classes e não terem feito uma análise mais detalhada ao nível dos métodos e dos atributos das mesmas. Todavia fizeram uma descrição mais detalhada no relatório. Achei o diagrama de classes incompleto dai qualificar o UML como fraco. Nesta fase não me parece que seja suficiente para que um engenheiro de software implemente este sistema.</p>	<p>grupoav.: 18 sol.....: Bom uml.....: Bom coment....: Falta uma descrição mais detalhada das classes e da relação com os use cases.</p>
<p>19</p>	<p>grupoav.: 19 sol.....: Bom uml.....: Bom coment....: O UML está de acordo com o que foi pedido nesta fase, simples e funcional. Tendo o relatório a explicação de cada classe, ficando bastante completa a solução.</p>	<p>grupoav.: 19 sol.....: Bom uml.....: Bom coment....: Tanto a qualidade da solução como o UML e o relatório parecem-me muito bons, o diagrama de classes está claro, daí a boa classificação.</p>

Sheet1

<p>grupoav.: 17 sol.....: Suficiente uml.....: Suficiente coment....:</p>	<p>grupoav.: 17 sol.....: Suficiente uml.....: Suficiente coment...: Na Classe PlanoDia na minha opinião seria talvez mais interessante implemetar as tarefas de um dia como uma hasmap em vez de um vector. Da forma que está definido (vector com 24 posições que equivale as 24h de um dia) delimita o tempo da tarefa a 1h, ou seja, não conseguimos definir mais de 24 tarefas para 1 dia. Já com a definição de hashmap o utilizador poderia inserir as tarefas que ele queira. Acho que a tarefa fixa também carece de uma duração, visto que esta também dura um determinado tempo, ou então a definição de data_inicio e data_fim inclua a hora_inicio e hora_fim.</p>	
<p>grupoav.: 18 sol.....: Bom uml.....: Bom coment...: De uma forma geral a solução apresentada aborda todos os aspectos necessarios a um gestor de tarefas que pretenda ir ao encontro das necessidades do utilizador. O relatório está muito completo e bem estruturado.</p>	<p>grupoav.: 18 sol.....: Suficiente uml.....: Bom coment...: Achei que o relatório poderia ser mais explícito em relação à função pretendida de algumas classes. Não percebi muito bem o conceito e o porquê dos dois "packages". Fiquei com algumas dúvidas que o relatório não explica. Por exemplo: a classe Lembrete é sub-classe da classe Tarefa. Estando a classe Lembrete ao mesmo nível das classes "Tarefa Fixa" e "Tarefa Móvel", isso quer dizer que apenas a classe Tarefa pode ter um Lembrete mas as classes "Tarefa Fixa" e "Tarefa Móvel" não podem ter Lembretes? Em relação ao UML, acho que ele está adequado à Solução Proposta mas tenho algumas dúvidas (sem certezas) quanto à necssidade de tantas ligações à classe Data/hora.</p>	<p>grupoav.: 18 sol.....: Bom uml.....: Bom coment...:</p>
<p>grupoav.: 19 sol.....: Suficiente uml.....: Bom coment...: A descrição das classes encontra-se um pouco vaga. Nota-se a falta de campos como a data de agenda, ou os prazos, que apesar de terem sido previstos na primeira fase do trabalho, não se encontram decritos nos atributos das classes. A ligação do interface ao código do sistema deveria ser feita em classe separada da classe principal de Gestão de Tarefas.</p>		

Sheet1

20	<p>grupoav.: 20 sol.....: Bom uml.....: Suficiente coment...: A explicação inicial está boa, mas a modelação arquitectural do sistema está algo incompleta. Alguns pormenores deveriam ter sido refinados.</p>	<p>grupoav.: 20 sol.....: Bom uml.....: Bom coment...: A solução apresentada é simples, mas cumpridora daquilo que se esperava para esta fase. Todo o trabalho criativo já havia sido desenvolvido na primeira etapa e sendo assim, nesta segunda etapa apenas era pedido que se estruturasse as relações entre as classes (e também os métodos), de uma maneira coerente com o trabalho já desenvolvido. Foi exactamente o que este grupo fez, por isso fica então a avaliação positiva.</p>
21		
22	<p>grupoav.: 22 sol.....: Bom uml.....: Bom coment...:</p>	<p>grupoav.: 22 sol.....: Bom uml.....: Muito Bom coment...: uma boa solução que resolve o problema proposto. A solução UML tem uma boa qualidade, analisando todas as classes previstas sobre uma boa solução para implementação.</p>
23	<p>grupoav.: 23 sol.....: Bom uml.....: Bom coment...: mais uma vez nesta fase, penso que o trabalho apresentado corresponde as expectativas, bem como aos objectivos desta cadeira</p>	<p>grupoav.: 23 sol.....: Muito Bom uml.....: Muito Bom coment...:</p>

Sheet1

<p>grupoav.: 20 sol.....: Bom uml.....: Bom coment....:</p>		
<p>grupoav.: 22 sol.....: Bom uml.....: Muito Bom coment....: O grupo apresenta uma solução ajustada às decisões delineadas na primeira fase. A complexidade do diagrama de classes é justificada convenientemente; ainda assim, creio que essa complexidade é introduzida por uma abordagem pouco prática no que concerne à separação dos conceitos relativos à agenda. Faz-se uma utilização diversificada dos relacionamentos possíveis entre as classes da aplicação (agregação vs. composição, multiplicidades), o que enriquece o diagrama criado e, conseqüentemente, a solução. Finalmente, o grupo conclui de forma interessante, relativamente ao desenvolvimento desta fase do trabalho.</p>	<p>grupoav.: 22 sol.....: Bom uml.....: Bom coment....: Descrição: de um modo geral, o que pretendem implementar está bem descrito. Sugeria apenas que pensassem no facto de que um utilizador pode não poder trabalhar 24h por dia e na possível existencia de dependencias entre as tarefas.</p> <p>UML: Penso que o actor do sistema não devia ser incluído(esta fora da fronteira do sistema) como classe, uma vez que o que estamos a modelar é o sistema.</p> <p>Ainda no diagrama de classe é dito que o sistema implementa a API, logo os cabeçalhos dos métodos que estão na API deveriam estar também na classe sistema, para posteriormente serem aqui implementados.</p> <p>Fica-se em duvida para saber como será despoletado um alarme. Devia existir uma thread, por exemplo, para estar em constante avaliação.</p>	<p>grupoav.: 22 sol.....: Muito Bom uml.....: Muito Bom coment....: Trabalho muito bem conseguido... excelente qualidade do relatório. Nitidamente trabalhado e estudado. Implementação pareceu-me boa também embora como óbvio havendo outras possibilidades talvez houvessem pequenas arestas a limar mas coisas ínfimas o k no global parece-me extremamente bem conseguido....</p>
<p>grupoav.: 23 sol.....: Bom uml.....: Suficiente coment....: Tem poucas classes e poucos métodos definidos.</p>	<p>grupoav.: 23 sol.....: Bom uml.....: Suficiente coment....: Embora a solução esteja boa, acho que poderiam ter aprofundado mais um bocão os métodos e mais classes. O UML podia estar mais rico... O relatório está BOM.</p>	

Sheet1

24	<p>grupoav.: 24 sol.....: Suficiente uml.....: Suficiente coment....: O diagrama de classes está bem estruturado, mas existindo o diagrama de estados da interface, tinha de existir a "interface" no diagrama de classes. Penso que essa é uma falha grave, que determina a avaliação.</p>	<p>grupoav.: 24 sol.....: Bom uml.....: Bom coment....: Em geral o trabalho está bom. Apenas não concordo com o atributo "Dependências" da class "Tarefa Movei", talvez por não perceber o que o grupo pretende com ele, e com os fragmentos da mesma, cada fragmento pode ser considerado uma nova tarefa.</p>
----	---	---

Sheet1

<p>grupoav...: 24 sol.....: Fraco uml.....: Fraco coment...: A solução visa pontos irrelevantes, tais como classe Utilizador, esta não é necessária uma vez que não existe autenticação de utilizador, existe apenas um utilizador por agenda. A classe tarefa Fragmentada 'e tambem desnecessaria uma vez que estar iria cair na classe das tarefas dependentes.</p> <p>A solução não preve escalonamento de tarefas uma vez que não existe nenhuma classe ou função com responsabilidade para tal. A solução não preve tambem procura de slots livres tal como na primeira fase. Não estão previstos conjuntos de tarefas. Ou seja a arquitectura é insuficiente.</p> <p>A qualidade do UML é tambem fraca uma vez que não existem agregações ou composições que em alguns dos casos seria necessaria. Não preve a existencia de construtores. Nem diferencia sequer dependencias de propriedades.</p>	<p>grupoav...: 24 sol.....: Bom uml.....: Bom coment...: Os diagramas de classes apresentados descrevem o que era pedido para esta fase do trabalho.</p>	
--	--	--

Sheet1

<p>25</p>	<p>grupoav.: 25 sol.....: Suficiente uml.....: Suficiente coment...: # Modelo # De acordo com o modelo apresentado na fase 1, penso que o diagrama de classes não contempla todos os casos de uso desenvolvidos. Por exemplo, nem todos os use cases que derivam do consultar agenda são contemplados neste diagrama de classes, dado que apenas existem o método na API que lista as tarefas diárias. Poderá deduzir-se daí que não é possível consultar as tarefas semanais, mensais e anuais. Resumindo, concluo que o diagrama de classes apresentado, não é fiel ao modelo apresentado na 1ªFase.</p> <p>#UML# Ao nível do UML, penso que o diagrama de classes não apresenta correctamente todos os relacionamentos entre todos os objectos. No caso, por exemplo da classe Data, esta relaciona-se com as classes Tarefa, Tarefa Fixa e Tarefa Móvel, contudo lembremo-nos que as setas a tracejado significam que a classe Origem implementa os métodos da classe Destino. Ora, será que querem mesmo dizer que a classe Data implementa os métodos das classes a que se liga, ou que, são essas classes que têm como atributo um objecto que é instância da classe Data? Outro aspecto, que embora não é muito importante se deve destacar, é o facto de que, dado que a classe Tarefa é superclasse das classes Tarefa Fixa e Tarefa Móvel, então estas herdam todos os métodos e atributos da sua superclasse, pelo que não seria necessário relacionar estas duas classes com a classe Data.</p>	<p>grupoav.: 25 sol.....: Bom uml.....: Bom coment...: O relatório apresentado está sucinto, contento toda a informação necessária para o entendimento da solução proposta.</p>
-----------	--	--

Sheet1

<p>grupoav...: 25 sol.....: Fraco uml.....: Muito Fraco coment...: Apenas algumas considerações: -Relatório muito mal estruturado e com alguns erros ortográficos. -O diagrama de classes tem várias inconsistências em relação ao diagrama de use cases apresentado na 1ª fase. -No diagrama de classes a relação entre classes é inadequada e muito pouco consensual. Veja-se, por exemplo, a associação entre a classe Alarme e a classe Tarefa (entre outros!). -Porquê nomear certas associações de "Extends" no diagrama de classes!?</p>	<p>grupoav...: 25 sol.....: Fraco uml.....: Fraco coment...: Supondo que a classe "Agenda" implementa a interface "API", porque tal não é explícito na descrição da classe, faltam os métodos da API na classe Agenda.</p> <p>Não percebo bem, como se guardam os alarmes, no GestorAlarmes temos o método inserirAlarme(alarme) que recebe a instância alarme, mas suponho que esse alarme devia ser inserido na Tarefa respectiva porque são as tarefas que contêm o atributo Alarme, como é que isso é possível?</p> <p>Pressumo que exista um ciclo esquisito usando setas de associação entre Agenda, GestorAlarmes, Alarmes e Tarefa.</p> <p>Não percebo o atributo -tarfMens: Vector na classe Agenda</p> <p>A classe ListaTarefasDiaria tem um vector de tarefas mas nenhum método que receba ou que crie tarefas, não percebo bem a utilidade desta classe e o seu relacionamento com as restantes.</p> <p>Descrição das classes, muito curta.</p>	
--	--	--

Sheet1

<p>26</p>	<p>grupoav.: 26 sol.....: Suficiente uml.....: Bom coment...: Projecto :</p> <p>O relatório esta bem esquematizado. O projecto esta de acordo com os use cases descritos na fase anterior. O modelo de domino não esta bem descrito, pois deveria descrever o que é realmente uma tarefa, textualmente ou através de um diagrama isolado só com isso. Continua a não haver a idéia de alocação de tarefas por blocos, a realocação não esta muito explicita, mas parece-me que no mudar tarefa fazem isso...</p> <p>UML :</p> <p>O diagrama esta bem estruturado. Faltam os construtores das classes. Falta uma classe API com os métodos que estarão visíveis ao utilizador. Na tarefa para ir de encontro com os use cases falta um atributo para a data, pois vocês inserem data de inicio e fim.</p>	<p>grupoav.: 26 sol.....: Suficiente uml.....: Fraco coment...: Aspectos Negativos:</p> <ul style="list-style-type: none"> - Não existe uma clara distinção das datas das tarefas (Falta-lhe uma distinção por anos). - Não faz referência as Multiplicidades (Por exemplo, uma agenda pode conter várias semanas, e uma semana vários dias 1->*). - Falta um aspecto importante, que são os pré-requisitos das tarefas; - Talvez não fosse necessário uma descrição da utilização do UML, bastava o porquê do que fizeram. - Falta de uma API - Gestor de tarefas continua com poucas opções. <p>Aspectos Positivos:</p> <ul style="list-style-type: none"> - Uma boa documentação das classes e métodos que resolveram usar. <p>Em resumo, a solução embora minimalista é apresentada de forma simples, mas com um UML com muitas falhas.</p>
<p>27</p>	<p>grupoav.: 27 sol.....: Bom uml.....: Bom coment...: Evolucao positiva da primeira fase para a segunda</p>	<p>grupoav.: 27 sol.....: Suficiente uml.....: Suficiente coment...: Não é implementado qualquer forma de recorrência nas tarefas.</p> <p>O realocar tarefa é definido na classe Tarefa é herdado pela Tarefa_Fixa e Tarefa_Movel, sendo que não é possível que o algoritmo usado neste método seja igual para ambos os tipos de tarefas.</p> <p>Nada é dito de como a Lista de dependencias na classe Tarefa_Movel é construida.</p> <p>As variáveis estado e prioridade poderiam ser inteiros, aumentando assim a eficácia no armazenamento, sendo a sua transformação em Strings significativas para o contexto deixadas ao cuidado da UI, por exemplo.</p> <p>A seta de Tarefa para a Agenda encontra-se com o sentido errado, pois sendo a Agenda que "usa" a Tarefa, a seta seria ao contrário.</p> <p>A seta do Alarme para a Tarefa também deveria ser ao contrário, pela mesma razão.</p>

Sheet1

<p>grupoav...: 26 sol.....: Suficiente uml.....: Bom coment...: A solução apresenta ainda coisas a melhorar .. Por exemplo : COmo é que a solução escalona as tarefas, não deveria existir uma classe escalonador para tratar disso ?</p> <p>Em termos de diagrama de classes no UML, está simples mas adequada à solução(que deve ser melhorada) só não existe a multiplicidade nas associações entre classes.</p>		
<p>grupoav...: 27 sol.....: Bom uml.....: Suficiente coment...: Podia ter uma classe responsável pelo tempo, ou seja a classe que iria dizer por exemplo se era possível ou não alocar determinada tarefa.Pois com esta classe, a classe agenda ficava mais fácil de implementar e as classes ficavam com uma estrutura mais organizada. Apesar de na qualidade da solução não se referirem, que o programa devia ter a interface serializable, no diagrama de classes devia estar, pois vai ser a interface que irá permitir guardar os dados no disco.</p>	<p>grupoav...: 27 sol.....: Bom uml.....: Suficiente coment...: A solução proposta apresenta uma boa arquitetura para o sistema de gestão de tarefas. Relativamente ao UML, o diagrama de classes apresentado, ilustra a modelação arquitectural proposta, no entanto, as setas de associação entre as classes Tarefa e Agenda e entre as classes Aviso e Tarefa, não foram utilizadas devidamente, ou seja, de acordo com a solução proposta (explicada na descrição textual das classes).</p>	

Sheet1

<p>28</p>	<p>grupoav.: 28 sol.....: Bom uml.....: Bom coment...: Há melhorias significativas relativamente à solução apresentada na 1a fase, sendo agora de notar uma nova coerência. O único reparo, e este relativamente à classe DataHora, seria de representar esta informação usando classes Java já existentes para este efeito. De resto, a especificação parece-me bem conseguida, e os diagramas de UML conseguem transmitir uma ideia correcta da descrição textual.</p>	<p>grupoav.: 28 sol.....: Fraco uml.....: Suficiente coment....:</p>
<p>29</p>	<p>grupoav.: 29 sol.....: Suficiente uml.....: Suficiente coment...: O projecto em geral está bom mas não usaram um processo iterativo para dos use cases extrair o diagrama de classes correspondente, apresentando apenas a versão final deste e não como lá chegaram. Em relação ao UML não utilizaram a notação sugerida para os maps, e deveria haver uma especialização da classe tarefa, uma vez que as tarefas fixas, móveis e flexíveis tem características diferentes que deveriam constar no diagrama. Têm um relatório curto mas conciso. A solução apresentada está de acordo com os use cases apresentados na fase 1.</p>	<p>grupoav.: 29 sol.....: Bom uml.....: Bom coment....:</p>
<p>30</p>	<p>grupoav.: 30 sol.....: Bom uml.....: Bom coment...: De uma forma geral, o trabalho parece cumprir os requisitos exigidos, de uma forma simples.</p>	<p>grupoav.: 30 sol.....: Suficiente uml.....: Suficiente coment...: Deviam ter pensado mais um pouco nas responsabilidades de cada classe (uma vez que, em quase todas as classes, só têm os atributos e os construtores), começando por não omitir os gets e sets (opção do grupo) que lhes facilitariam já o trabalho para a terceira etapa. Deste modo, também para os clone e toString. Um pormenor: os nomes dos métodos deviam começar por minúscula e apenas os construtores é que começam por maiúscula. Deviam ter explicado melhor para que vai ser a classe Scheduler, pois praticamente apenas dizem que vão precisar dela. À qualidade da solução atribui Suficiente porque não possuem histórico e porque deviam ser mais claros em alguns pormenores. Por exemplo, os avisos vão ser guardados num Map com uma chave String e um valor do tipo Tarefa, que suponho seja outro Map diferente do Map das tarefas e do das tarefasAtrasadas como me pude aperceber, mas apenas, na classe GestorTarefas.</p>

Sheet1

<p>grupoav.: 28 sol.....: Suficiente uml.....: Suficiente coment...: Uma das maiores falhas é que não é mencionado como os alarmes das tarefas serão tratados. Não é referido, também, como são tratadas as tarefas que podem ser fragmentadas (tarefas não fixa).</p>	<p>grupoav.: 28 sol.....: Fraco uml.....: Suficiente coment...:</p>	<p>grupoav.: 28 sol.....: Suficiente uml.....: Suficiente coment...: Não foi feito o diagrama de estados.</p>
<p>grupoav.: 29 sol.....: Suficiente uml.....: Suficiente coment...: Penso que o minimo foi realizado, mas não encontrei a API para a interface com o utilizador, o que não deixa de ser grave. Penso porém que podiam ter criado uma classe abstracta Tarefa e daí criar três sub-classes, Fixa, Flexível e Móvel. Mas isso são opiniões... O Relatório estava razoável também..</p>		
<p>grupoav.: 30 sol.....: Muito Bom uml.....: Bom coment...: Apesar de estar planeada a sua descrição, em desenvolvimentos futuros do trabalho, convinha precisar um pouco mais qual a função da classe "Scheduler" para que não restassem dúvidas nesse aspecto. Tirando esse ponto, uma solução muito boa.</p>		

Sheet1

<p>31</p>	<p>grupoav.: 31 sol.....: Bom uml.....: Bom coment....: Os métodos get's e set's não eram necessários colocar no Diagrama de classes.</p> <p>Quanto ao método guardar(Ficheiro) acho que é importante e que nem todas as pessoas se devem ter lembrado.</p> <p>Em relação ao relatório, penso que a ideia do professor era fazer um relatório em todas as fases e não aproveitar o relatório da fase anterior acrescentando apenas o diagrama de classes.</p>	<p>grupoav.: 31 sol.....: Muito Bom uml.....: Suficiente coment....: Falta agregação entre gestor e tarefa</p>
<p>32</p>	<p>grupoav.: 32 sol.....: Muito Fraco uml.....: Muito Fraco coment....: O ficheiro F2_DSI.zip tem 0 bytes, por isso não posso avaliar o trabalho dos meus colegas.</p> <p>(O único outro ficheiro é o do trabalho 1.)</p>	<p>n. grupo que avalia: 32</p> <p>Qualidade Solucao : Bom</p> <p>Qualidade UML : Bom</p> <p>Comentario: Acho que a solução esta realmente bastante boa. Quanto ao relatorio esta muito perceptivel devido a inclusao dos diagramas de actividades, o que da facilmente uma ideia geral do trabalho. Esta claro e conciso.</p>

Sheet1

<p>grupoav.: 31 sol.....: Bom uml.....: Bom coment...: Funcionalidades apresentadas bastante boas, e opções disponíveis muito bem documentadas. Variedade de "avisos" disponíveis. Considera-se uma boa solução, bem documentada e estruturada.</p>	<p>grupoav.: 31 sol.....: Bom uml.....: Bom coment...:</p>	

Sheet1

33	<p>grupoav.: 33 sol.....: Bom uml.....: Muito Bom coment...: UML:</p> <p>Parece-me bem.</p> <p>Solução:</p> <p>Pelo que entendi, a classe "tarefa fixa", pode ser atribuída para tarefas fixas e tarefas moveis (o que o nome da classe "tarefa fixa" não é muito feliz). Sendo este o caso, se a variável "flexivel" fosse 'true', ou seja, se a tarefa fosse movel, seria necessário mais duas datas, duas dadas pelo utilizador que identifica a data inicial e final possíveis para a alocação da tarefa e outras duas que seriam as datas "verdadeiras" alocadas pelo gestor para a respectiva tarefa. Só com duas datas como resolvem esta situação??!</p> <p>Não elaboraram um package com a interface gráfica, GUI, não identificando as restantes classes num package diferente.</p> <p>Quanto ao resto, parece-me bem.</p>	<p>grupoav.: 33 sol.....: Suficiente uml.....: Suficiente coment...: A solução pareceu-me um pouco confusa. Quanto ao UML, no diagrama de classes faltam algumas relações entre as diversas classes.</p>
----	--	--

Sheet1

<p>grupoav...: 33 sol.....: Suficiente uml.....: Suficiente coment...: Em relação à solução: Satisfatória. - É positivo ver a introdução (em relação à fase 1) de um índice (ainda que sem título), mas as páginas deviam estar numeradas ao longo do relatório... - Também foi adicionada uma conclusão, - ... mas continua a não haver bibliografia. - Continua-se a verificar ausência de data no relatório - O texto não está justificado na descrição detalhada das classes.</p> <p>Em relação à qualidade do UML: Satisfatório - Nem a classe Agenda nem a classe Notificador implementam totalmente a interface AgendaAPI... - A notação do UML (as setas entre as classes) implicam que não é necessário representar essas mesmas variáveis nas classes - A classe TarefaRecorrente tem "todas" as 12 variáveis de classe da classe TarefaFixa. Porque não captar esta repetição com o mecanismo de herança da classe abstracta Tarefa? - É uma boa ideia utilizar as classes de Java Thread e Observable. - Qual é o domínio do map presente na classe Agenda? Suponho que seja a data (sendo o contra-domínio AgendaDiaria), mas isto não é referido, e também existe uma notação própria de UML para maps. - Continua a haver redundância nas duas listas de dependências e continuo a achar que a opção de utilização de redundância não está devidamente justificada. - Não foi resolvido o problema de existirem tarefas com múltiplos slots alocados para a sua realização (existe uma variável booleana particionavel, mas isto não representa as datas de início e fim dos múltiplos slots)</p>	<p>grupoav...: 33 sol.....: Bom uml.....: Bom coment...: A solução encontrada ia de encontro ao esperado. O UML descrito define todas as classes necessárias à solução proposta. No entanto peca por demasiado complicada, e com pormenores que considero desnecessários, estando no entanto correctamente especificados. O relatório é que poderia estar melhor, mais organizado e com mais informação sobre o diagrama de classes.</p>	
---	---	--

Sheet1

<p>34</p>	<p>grupoav.: 34 sol.....: Bom uml.....: Suficiente coment...: Acho que o diagrama de classes está nesta fase esclarecedor e satisfaz razoavelmente o objectivo proposto. Tanto quanto sei, o relatório deve corresponder a esta fase do trabalho sendo portanto negativo iniciar secções que não estão completas por virem a ser completadas. Ou não as iniciavam, ou completavam-nas. De um modo geral, a solução proposta está boa se bem que o diagrama de classes me pareça um tanto incompleto (o que considero normal, visto estar a ser completado por fases. Tenho a certeza que na próxima avaliação verei um diagrama de classes bastante diferente). Como já disse existe ainda o pequeno problema da organização do relatório, se bem que de resto está muito bom.</p>	<p>grupoav.: 34 sol.....: Bom uml.....: Bom coment....:</p>
<p>35</p>	<p>grupoav.: 35 sol.....: Suficiente uml.....: Suficiente coment...: Esta fase do trabalho sofreu uma evolução positiva perante a fase anterior. O UML e a solução estão melhores. Relativamente ao diagrama de classes, grande parte dos seus métodos são set's e get's e existem variáveis com o mesmo significado(ex: as variáveis grau e descrição na classe prioridade). Numa fase posterior estes aspectos deverão ser melhorados. O relatório deverá ser enriquecido e mais completo.</p>	<p>grupoav.: 35 sol.....: Suficiente uml.....: Suficiente coment...: O diagrama de classes está de acordo com o trabalho realizado na fase anterior. .Falta da noção de lógica de negócio. .Falta a classe de gestão de tarefas e respectiva interface(API). .Falta a classe GUI (grafic user interface). .Relações entre classes mal implementadas.. .Apenas a relação entre as classes tarefa e tarefa com Alarma está correcta pois representa uma generalização. Contudo, a implementação continua bastante simples...</p>
<p>36</p>	<p>grupoav.: 36 sol.....: Bom uml.....: Bom coment...: De uma forma geral a solução apresentada está bastante satisfatória. O UML também está bom; falta um pequeno pormenor, que era dividir a solução em dois packages, como o professor sugeriu na aula teórica.</p>	<p>grupoav.: 36 sol.....: Bom uml.....: Bom coment....:</p>

<p>grupoav...: 35 sol.....: Bom uml.....: Bom coment...: Apesar do relatório apresentado estar sucinto, este contém toda a informação necessária para o compreendermos a solução proposta.</p>		
<p>grupoav...: 36 sol.....: Bom uml.....: Bom coment...: Não existe nenhuma variável na Classe Agenda para guardar os agendamentos! A variável rep devia ser um enumerável. Por várias vezes são referidos parâmetros de métodos, mas nenhum desses métodos tem os parâmetros descritos na sua assinatura (tanto na descrição como no UML). Os métodos criaDep e limpaDep aparecem na descrição mas não no Diagrama de Classes. A classe agenda aparece como abstracta na descrição, mas não no UML.</p>		

Sheet1

37	<p>grupoav.: 37 sol.....: Suficiente uml.....: Suficiente coment...: Pouca evolução relativamente ao ultimo relatório. A diferença entre os diagramas de classes do primeiro relatório e este são mínimas! Nenhuma das Classes possui métodos, que era o pedido nesta fase.</p>	<p>grupoav.: 37 sol.....: Suficiente uml.....: Suficiente coment...: Nesta segunda fase o grupo apresentou uma solução de acordo com a solução que já tinha apresentado na 1ª fase do projecto , com pontos em que podiam ter melhorado como a elaboração do relatório , já que se trata de um relatório muito pouco elaborado , facto que deviam tentar melhorar para a próxima fase . No entanto quanto à qualidade do UML achei que de acordo com a solução apresentada é boa . Na generalidade conseguiram os objectivos principais desta fase que era uma descrição da arquitectura com base nos use cases do projecto da 1ª fase.</p>
38		
39	<p>grupoav.: 39 sol.....: Suficiente uml.....: Bom coment...: A solução apresentada é muito simplista não existe qualquer referencia a prioridades no sistema de alocação de tarefas, sendo esta uma característica fundamental para o sistema. Para além disto também o escalonamento apenas pode ser feito de semana a semana, não permitindo isto tarefas que sejam repetidas semanalmente, que a sua duração seja maior do que uma semana ou ainda um planeamento antecipado da semana seguinte. Em relação ao UML é apresentado o diagrama de classe com a identificação dos objectos envolvidos no sistema e os métodos a eles inerentes. Contudo não existe qualquer referencia à multiplicidade existente entre objectos nem à composição ou agregação entre as classes.</p>	<p>grupoav.: 39 sol.....: Suficiente uml.....: Suficiente coment...:</p>

Sheet1

<p>grupoav...: 37 sol.....: Bom uml.....: Muito Bom coment....: uma solução simples, mas parece eficaz.</p>		
<p>grupoav...: 39 sol.....: Suficiente uml.....: Suficiente coment....: Peca bastante por apenas permitir agendar tarefas para no maximo dai a uma semana</p>	<p>grupoav...: 39 sol.....: Muito Fraco uml.....: Muito Fraco coment....: O diagrama de classes parece-me muito incompleto, pois modela pouco mais que o domínio do problema. Esperava-se mais aprofundamento do comportamento do sistema nesta fase, como métodos necessários nas classes, mais informação nos métodos existente (nenhum dos métodos especifica parametros ou valores de retorno) e possivelmente entidades de "gestão" (embora exista a classe 'scheduler' que será a classe principal do sistema). A utilização do UML eh perfeitamente básica, havendo até erros, como por exemplo na relação entre o alarme e a tarefa (deveria ser no sentido inverso), a referência às classes incluídas como variáveis de instância (redundante) ou a ausência de especificação da multiplicidade das relações. A qualidade da solução também me parece muito fraca. Não há quase nenhum acresceto ao que foi modelado na primeira fase, o que era requerido, ou pelo menos desejável. Não me parece também que haja ideias minimamente sólidas sobre o comportamento do sistema (o que origina um diagrama UML muito incompleto).</p>	