Planning Module Report

1 Student deliver a file named “3DJ-065”

Deadline: November 24th 23:59

|  |  |  |  |
| --- | --- | --- | --- |
| Membros da equipa | | | |
| José Teixeira - 1210965 | Rodrigo Almeida - 1220776 | Rafael Santos - 1221267 | Alexandre Costa - 1211225 |

|  |  |  |
| --- | --- | --- |
| User Stories | | |
| 6.3.1 As an Admin, I want to obtain the better scheduling of a set of operations (surgeries) in a certain operation room in a specific day. The better scheduling is considered as the sequence of operations that finishes early. Note that surgeries have constraints (e.g. number of doctors or other staff), namely concerning the time availability of staff during the day. The approach may be generating all surgeries’ sequences and select the better, and this is possible till a certain dimension (number of surgeries). The user must have a user interface to start the process (enter any additional parameters the planning algorithm needs, e.g., room number, date). The system will then generate the plan and show it to the user on the screen. It is acceptable that the UI blocks while waiting for the planning module response. | 6.3.2 As an Admin, I want to know till what dimension in terms of number of surgeries is possible to ask for the better solution. Perform a complexity analysis of the problem to understand to which dimension it is feasible to ask for the better solution. Document your results and findings. | 6.3.3 As an Admin, I want to obtain a good schedule, not necessarily the better, in useful time to be adopted. The system generates a "good" (non-optimal but efficient) schedule using heuristics or informed methods (e.g., greedy algorithms, rule-based scheduling). The system prioritizes generating a schedule that is close to optimal but does so within a reasonable time frame (e.g., under 30 seconds). The user must have a user interface to start the process (enter any additional parameters the planning algorithm needs, e.g., room number, date, which heuristic to use). The system will then generate the plan and show it to the user on the screen. It is acceptable that the UI blocks while waiting for the planning module response. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Points to cover | | | | |
| **1)** Brief explanation of the basic code to schedule just considering doctors (the part presented in the first 3 TP classes) – 10% | **2)** Complexity Study (6.3.2; just with doctors)– 15% | **3)** Two Heuristics, including the comparison with the better solution (6.3.3)–30% | **4)** Adaptation of the method to include all necessary staff and all phases of the operation (6.3.1). – 35% | **5)** Conclusions - 10% |

1. Explicação do Código base(considerando apenas doutores)

|  |  |  |  |
| --- | --- | --- | --- |
| Predicado | Argumentos | Exemplo | Explicação |
| staff | (StaffID,  Role,  Specialization, CanPerformSurgeryType) |  |  |
| agenda\_staff | () |  |  |
| timetable | () |  |  |
| surgery | () |  |  |
| surgery\_id | () |  |  |
| assignment\_surgery | () |  |  |
| find\_free\_agendas | () |  |  |
| free\_agenda\_staff0 | () |  |  |
| free\_agenda\_staff1 | () |  |  |
| adapt\_timetable | () |  |  |
| treatin | () |  |  |
| treatfin | () |  |  |
| intersect\_all\_agendas | () |  |  |
| availability | () |  |  |
| intersect\_2\_agendas | () |  |  |
| intersect\_availability | () |  |  |
| min\_max | () |  |  |
| agenda\_operation\_room | () |  |  |
| obtain\_better\_sol | () |  |  |
| availability\_all\_surgeries | () |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| findall | () |  |  |

2)

3)

4)

5)