# Sprint Plan # 7

Context Project: Health Informatics
Group: HI2 / group B

| User Story # | Task # | Priority** | Task Assigned To* | Estimated Effort per Person per Task (in hours) |
|---|---|---|---|---|
| | Emergent Architecture Design | 2 | Rick, Arnout | 2, 2 |
| | Test InputPage, InterResultController, Help package | 1 | Pascal | 5 |
| | Test Chunk, Interpreter, Parser package | 1 | Wim | 5 |
| | Test Connection, Task, Constraints, XMLDocument | 1 | Rick | 5 |
| | Test interfacecomponents package | 1 | Arnout | 5 |
| Analysts wants to make graphs of the Result data and export. | Output Page write results to file | 2 | Jelmer, Wim | 2, 2 |
| | Output Page checkboxen for which graphs | 2 | Wim, Jelmer | 2, 2 |
| | Output Page frequency bar | 2 | Jelmer, Wim | 2, 2 |
| | Output Page State-transition matrix | 2 | Wim, Jelmer | 2, 2 |
| | Output Page Boxplot | 2 | Wim, Jelmer | 2, 2 |
| | Output Page stam and leaf | 3 | Wim, Jelmer | 2, 2 |
| | Output Page histogram | 4 | Jelmer, Wim | 2, 2 |
| | Output Page export graph | 3 | Jelmer, Wim | 2, 2 |
| Analysts wants to use a language to output data | Computation implementation | 2 | Arnout, Rick | 4, 4 |
| | Computation test | 3 | Pascal | 3 |
| | Conversion implementation | 2 | Rick, Pascal | 3, 3 |
| | Conversion test | 3 | Arnout | 2 |
| | Comparison implementation | 2 | Rick, Arnout | 4, 4 |

| | | | | |
|---|---|---|---|---|
| | Comparison test | 3 | Pascal | 3 |
| | Check gui on duplicated methods | 2 | Wim, Jelmer | 4, 4 |
| | Save, merge table pop-up button | 2 | Jelmer, Wim | 2, 2 |
| | Error handing analyse | 2 | Arnout, Rick | 2, 2 |
| | Data selection on line | 3 | Pascal, Arnout | 3, 3 |
| | Code page help update | 3 | Pascal, Rick | 2, 2 |
| | Save database | 2 | Arnout, Rick | 3, 3 |
| | Database on project | 2 | Arnout, Rick | 4, 4 |

* The first person in the Task Assigned To area is responsible for the Task.

** Scaled into 1 – 5 where 1 has the highest priority and 5 the lowest. Scale 4 and 5 could be passed on to a next sprint.

## Priority

Main core of this sprint is to get the product to a working version with all must haves. Before we continue with the project, all data that is added last sprint should be tested thoroughly, highest priority. The rest of the priority scalars are divided in 3 classes. The first class are the must-haves for the product, these tasks have a priority of 2. The second class are the should-haves of the product, these tasks have a priority of 3. The last one are the could-haves of the product, these tasks have a priority of 4. Priority 4 will be implemented, only if we have time left to do it.