# Sprint Plan 8

## Context Project: Health Informatics

Group: HI4

| User Story | Task | Task Assigned To | Estimated Effort per Task |
|---|---|---|---|
| The user wants to analyze multiple persons at the same time | - Implement the analysis of multiple persons with the output in a single file. | Remi | 5 Hours |
| The user wants to chunk on day of the week and on time of day | - Implement chunking on day of the week<br>- Implement chunking on time of the day | Hans<br>Hans | 2 Hours<br>2 Hours |
| The user wants to see all the timestamps in the results table (not in a column), for an overview. | - Add the timestamp to the results table. | Sven | 3 hours |
| The user wants to use the tool to answer questions about the behavior of patients. Recheck all the questions and find example solutions. | - What time of the day and on what day do people measure themselves?<br>- What time of the day and on what day do they enter the measurement in mijnnierinzicht? | Hans<br>Hans | 2 Hours<br>2 Hours |

| | | | |
|---|---|---|---|
| | - What is the difference in time between measurement device and entering data in Mijnnierinzicht? | Hans | 2 Hours |
| | - Is there a difference between StatSensor measurement and what patients enter into Mijnnierinzicht? | Matthijs | 2 Hours |
| | - How often do patients measure themselves before they enter data into Mijnnierinzicht? | Matthijs | 2 Hours |
| | - If a patient did measure multiple times, what measure do he/she eventually enter into Mijnnierinzicht? | Elvan | 2 Hours |
| | - Do patient follow up advice given my Mijnnierinzicht? | Hans | 2 Hours |
| | - What are the conditions under which people start deviating from their normal measurement routine? | Remi | 4 Hours |
| | - What are the conditions under which people overwrite their initial data entered in Mijnnierinzicht? | Remi | 4 Hours |
| | - Find cases where there is a difference between device measurement and what is entered in Mijnnierinzicht | Matthijs | 2 Hours |
| | - Find cases where Mijnnierinzicht advice to contact the hospital | Hans | 2 Hours |
| | - How well do patients follow up advice of Mijnnierinzicht to re-measure again? | Elvan | 2 Hours |
| | - Do external factors, such as holidays, have an effect on the patient's measurement routine? (partially answered by graphs) | Elvan | 3 Hours |

| | | | |
|---|---|---|---|
| | - Is the value of creatinine, blood pressure, HR affecting the patient's measurement routine? (partially answered by graphs) | Elvan | 3 Hours |
| The user wants to know how to use the program so that he can easily use it | - Add tooltips to the help pages.<br>- Create a file that describes how to use the program for a few examples<br>- Write some script examples and explanations for the operations in the user manual | Remi<br>Remi<br><br>Everyone | 1 Hour<br>2 Hours<br><br>1 Hour p.p. |
| The user wants to visualize the final graphs with a state transition matrix and Markov Chain graphs to have an overview of the data | - Create a state transition matrix<br>- Implement Markov Chain graphs<br>- Show multiple graphs, per chunk. | Matthijs<br>Matthijs<br>Matthijs | 3 hours<br>3 hours<br>4 hours |
| The user wants to have a timeline (time series) that visualises all the events happened during a period of time to explore the data visually | - Let the user specify a particular period of time<br>- Create a timeline that shows all the events that happened during that period of time | Sven<br>Sven | 2 Hours<br>3 Hours |
| The user wants the GUI to be user-friendly and complete | - Syntax highlighting for all constructs.<br>- Add COL() to the parsing of a column name | Remi<br>Elvan | 2 hours<br>2 hours |
| The user wants a functional software application that is easy to use | - Testing and improving important features<br>- Perform usability testing with at least five people | Everyone<br>Sven | 5 Hours pp<br>5 Hours |

| | | | |
|---|---|---|---|
| SIG and the user want the code to be clean, documented and well tested so it can easily be extended for future use. | - Refactor the codebase<br>- Make long methods shorter.<br>- Check PMD, Checkstyle and FindBugs and fix it.<br>- Write tests to improve coverage. (Import controller, and such) | Everyone<br>Everyone<br>Sven<br>Everyone | 2 hour p.p.<br>1 hour p.p.<br>2 hours<br>3 hour p.p. |
| The client wants a finished emergent architecture document | - Finish the architecture document (by adding design patterns, updating the subsystem decomposition and UML) | Elvan | 3 Hours |
| The client wants a draft of the final report. | - 1. Introduction, including a brief problem description and end-user's requirements<br>- 2. Overview of the developed and implemented software product<br>- 3. Reflection on the product and process from a software engineering perspective<br>- 4. Description of the developed functionalities<br>- 5. Special section on interaction design (development of the HCI module)<br>- 6. Evaluation of the functional modules and the product in its entirety, including the failure analysis<br>- 7. Outlook | Elvan (evaluation by Sven)<br>Elvan (evaluation by Sven)<br>Sven (evaluation by Matthijs)<br>Hans (evaluation by Elvan)<br>Matthijs (evaluation by Remi)<br>Remi (evaluation by Hans)<br>Remi (evaluation by Hans) | 2 Hours<br>2 Hours<br>4 Hours<br>4 Hours<br>3 Hours<br>3 Hours<br>1 Hour |

| | | | |
|---|---|---|---|
| The client wants to have a nice presentation that we can perform to the clients | - Create a presentation concept.<br>- (optional) Create an initial powerpoint presentation. | Everyone<br>Remi | 1 Hours p.p.<br>1 Hour |

## Remaining Tasks

(non-sprint related tasks that are not mentioned above)

- Acceptance testing with Wenxin Wang
  On Tuesday or Thursday we are going to meet with Wenxin to get her feedback on the workflow and graphical user interface of the program.in a one hour meeting.
- Extra room for flexibility
  We left 2.5 hours room for each group member for flexibility.
- 2-pager of project skills
  Every team member needs 2 hours time to write the 2 pager.

## General explanation on responsibilities

- Programming tasks
  The group member who is assigned to a programming task has the responsibility to implement the corresponding feature and to fully test it (with at least 75% line coverage). After a feature is done, he should open a pull request for it and make sure that the code is approved by at least two other group members and merged with the master branch before the deadline of the current sprint.
- Documenting tasks
  The group member who is assigned to a documenting task has the responsibility to write the corresponding sections and to perform a spell check. After the part is written, he should open a pull request for it and make sure that the document is approved by at least two other group members and that all sections are merged with the master branch before the deadline of the current sprint. If the document also has to be uploaded on Blackboard, he is responsible for doing this on time.
- Final Report
  For the final report the same responsibilities hold as for the general documenting tasks. The only difference is that two group members will be assigned to every subsection of the report instead of just one. One group member has the responsibility to write the subsection and the other one has to re-evaluate it before opening a pull request.