

Progress report

Third implementation phase

PMoC

24.3.2002

Version 1.1

T-76.115 Progress report – Project planning phase PMoC

Version	Date	Author	Description
1.1	24.03.03	JL	Proofreading. Fixed some errors.
1.0	24.03.03	JF, ML	

Table of Contents

1. Project status	3
1.1 Overview	3
1.2 Accomplished tasks	5
2. Work performed	6
2.1. Organizing and planning	6
2.2. Implementation	6
2.2.1 Implemented parts during this phase.	6
2.2.2 Half done parts	7
2.3. Design	7
2.4. Documentation	7
2.5. Peer testing	7
3. Applied process	8
3.1. Project management	8
3.2. Risk management	8
3.3. Version management	8
3.4. V-model testing	8
3.5. Meeting practices	8
3.6. Coding conventions	8
3.7. Integration and regular builds	8
3.8. Use-cases	9
3.9. Customer and mentor relations	9
4. Improvement	9

1. Project status

As the project has been a long term project that has required quite a lot of effort from the project group, a slight weariness has been evident in the group, and especially during the beginning of the I3 phase this was manifested as somewhat slower progress than the average. This has been a trend for our group more or less through all of the project, and in accordance with our ‘trends’ the activity has also grown better and better towards the end of the phase.

The overall progress has still been very significant and the software is evolving close to its final form.

1.1 Overview

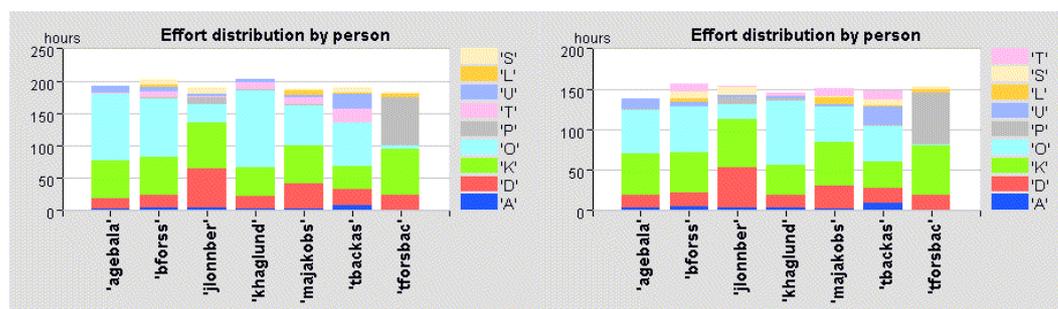


Figure 1 & 2 effort entire project at end of I3 phase vs. end of I2 phase

As can be seen from figure 1 above, after this phase the workload has been quite well shared. A positive thing to notice is that e.g. agebala and khaglund have worked very well compared to earlier phases (looking at the hours). The hours for tforsbac do not quite give the right picture as some of the heavy parts (planning the next phase, writing delivery documentation and preparing for the project review) is mainly done after this document has been written.

It can also be established that the budget for the project will be overrun slightly, but not as much as expected in a worst case scenario. *The budget for this project until the end of the I3 phase was approximately 1251 hours. The realized time was a total of 1288 hours.* Thus, we only have an overrun of 37 hours so far.

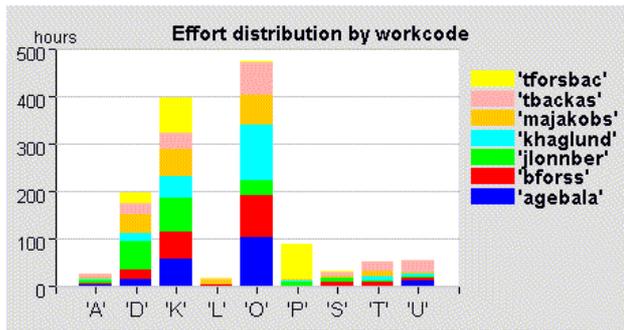


Figure 3 effort by WC, entire project end of I3

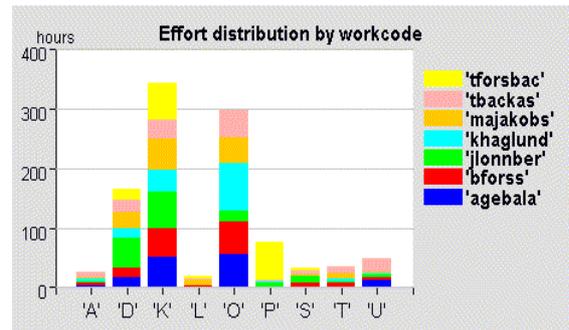


Figure 4 effort by WC, end of I2

From these two figures we can see that the hours spent on programming finally have grown bigger than the hours spent on meetings. Other parts have only grown slightly, quite in line with the development of the project.

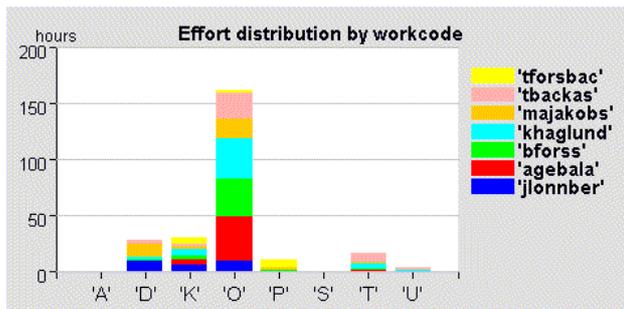


Figure 5 effort phase I3

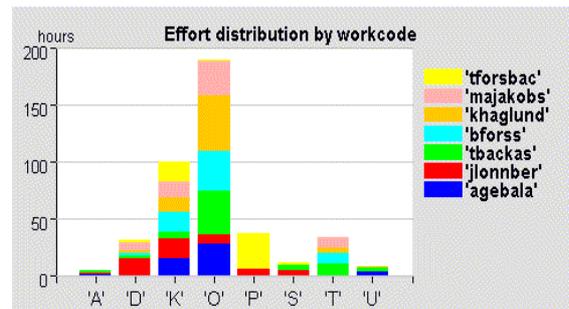


Figure 6 effort phase I2

Finally from figure 5 and 6 above we also see that all other parts have been minimized except for programming, which should only be considered natural and actually a very healthy sign at this stage of the project. The only other part which has remained as large as in the last phase is the documentation part, which partly is explained through the additional documentation required for the peer testing. The decrease of hours spent on testing is also partly explained by the advantage of peer testing, partly by the fact that testing has become more routine.

1.2 Accomplished tasks

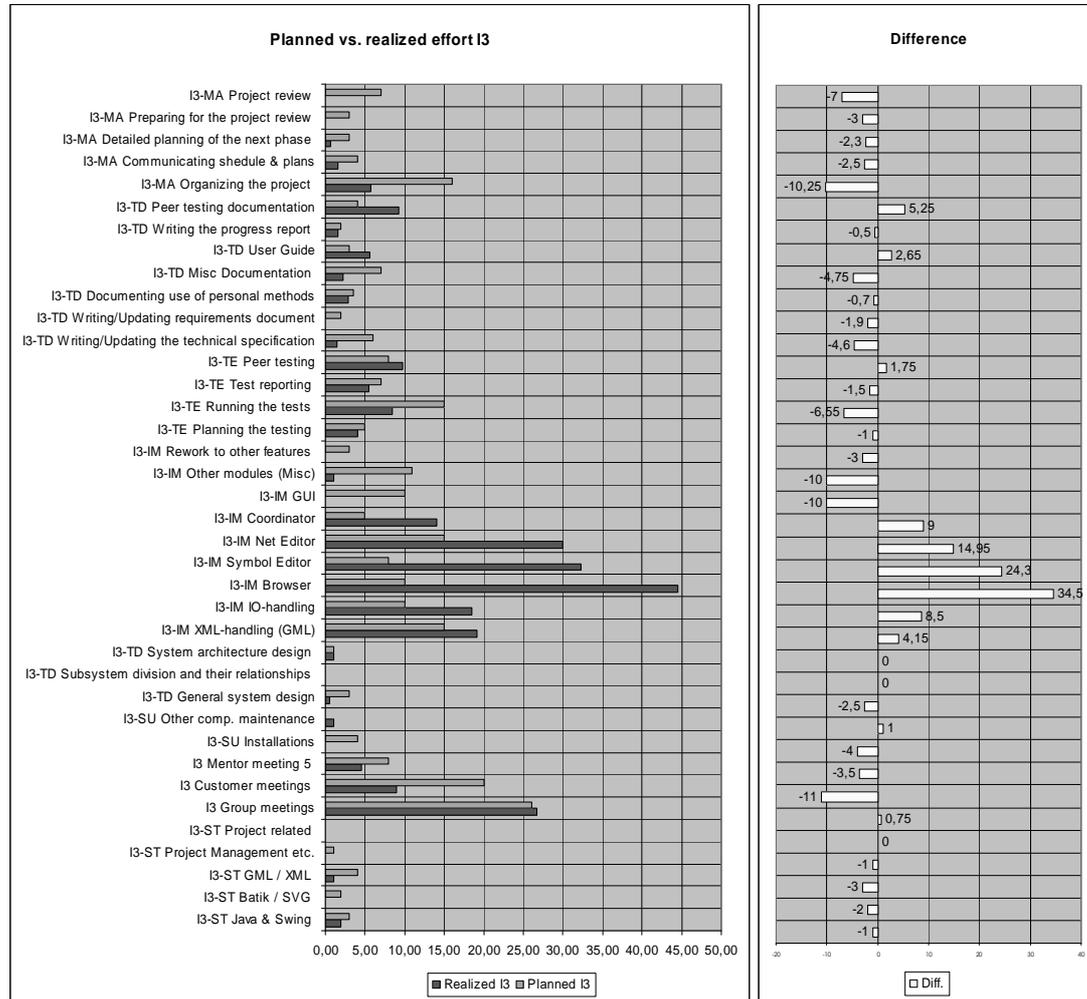


Figure 7 Planned vs. realized effort and difference to the plans

Comments:

Mainly, the implementation categories have been overrun, but most other parts have not required as much hours as estimated. So, as a whole, the project hasn't got out of hand, and still it has been possible to accomplish most of the required tasks. From this we can also conclude that, up to now the budget has allowed all other parts to be executed as required for this project, but that there should now yet be time for finishing the implementation well. Alternatively the implementers should have had a bigger budget during the last two phases, but in the context of this project that wouldn't have been possible as all group members should have an equal amount of 200 working hours.

2. Work performed

2.1. Organizing and planning

During this phase the group members have been able to mainly devote themselves to implementation. Also, thanks to well made specifications and design the implementation has been mostly straightforward from that point of view, but technical issues regarding the external packages used (e.g. Batik) have required a significant amount of time, to meet the specific needs of this project.

Less organizational effort has been needed than in the I2 phase.

2.2. Implementation

The implementation work proceeded at a good pace. After some of the features were dropped or simplified in agreement with the customer the remaining work was easily identified. The GML group was able to finish the saving of the GML structures and make the browser function as desired. The graphics group added tools for drawing connections and a terminal toolbar for adding terminals more easily. The synchronization of the browser and the canvases was finished and functionality added to right-click menus for nodes in the tree and graphical objects as specified. At the end of the phase, the system was functioning in most ways as required.

2.2.1 Implemented parts during this phase.¹

- The browser tree
 - Icons for different GML types in the tree
 - Right-click functionality for nodes
 - Dynamic loading of content
- Propagating events between browser and canvas
 - Select, delete and other functions propagated
 - Drag-and-drop support enhanced and visual feedback added
- IO-interface
 - Saving of GML structures
 - Saving and loading of nets correctly
- Graphics & GUI
 - Implementing connections between components
 - Connection look editor
 - Terminal tool bar for adding terminals
 - Right-click functionality for graphical objects
 - Changing properties for graphical object (dimensions, color...)

The features that are not yet working are few. Most of them are also simple to implement or already have an alternative implementation in the program. They have been left to the bottom of the TODO list as they are not vital for the system to work as required.

¹ Many software- and project specific terms are used. For explanations of these are found in following documents 1) User Requirements © PMoC 2) Technical specification © PMoC 3) GML-specifications © VTT. These can be found under the project web-page from the forum (requires login which is freely available) www.yawc.net/pmoc

2.2.2 Half done parts

- Tree browser
 - Editing property values
 - Opening new tree views from a specified node
 - Importing component types
- Graphics
 - Import of external graphics partly working
 - Connection editor was made very simple.
 - Saving connections symbols

2.3. Design

The design of the program has not changed very much during the phase, but certain features have been dropped or simplified in agreement with the customer.

- The terminal toolbar was designed
- Connections were designed

2.4. Documentation

The technical specification and user requirements documents have been updated to reflect the changes in design described above and the changes in required features necessitated by the design changes. In general, changes have been made to the technical specification first and then propagated to the use cases in the requirements document, as this matches the customer's way of expressing himself better.

The documents describing our software development practices have been updated to reflect our experiences from this phase. A test report² has been written that describes the results of the testing performed according to the test plan³, and a log file⁴ was also produced from the automated unit tests. A separate test plan was written for the peer group as well as a peer test report template. The user's guide was completed for use in the peer testing, although it was primarily written for the customer. A test report was also written for the peer group based on our testing of their system. As you no doubt have noticed, a progress report summarising the work done this phase has been written. The project plan has been revised to include plans for the last phase.

2.5. Peer testing

The peer testing for our system was conducted by OpenLogbook on week 10. The results were not very useful for our group, mostly due to the fact that the peer testers did not know how to use the most important functions of the program. The testers did not manage to instantiate components and form nets of components, which was the most important aspect of the peer testing, in spite of the user manual and test plan. Due to the complex nature of the project we were prepared for this and had advised the peer group to contact us (1.3 and risk management section in peer test plan) in case there were problems. Thus

² Markus Jakobsson/PMoC: Test Report for I3

³ Markus Jakobsson/PMoC: Test Plan

⁴ Markus Jakobsson/PMoC: Log file for I3

we were a bit surprised when the peer group was unable to execute some of the test and had not contacted us during the testing, even with an e-mail regarding the problems. We could easily have sorted out the problem. The result was that that some test cases were not executed. Our user guide was probably not as helpful for the testers as they had hoped, but our user guide is written for the customer's needs, and writing a beginners guide solely for peer testing was not possible with our budget.

Otherwise the test report was well written and contained useful comments from a more usability point of view and a few suggestions for improvements. The number of bugs found was surprisingly low however. To summarize, the testing of the peer group was less useful than our own testing and did not discover many important bugs in the system. This was however expected as peer testing concentrated more on usability while the customer's interest lies in the functionality of the system and the implementation of the GML structures. The testing of these aspects with peer testing would not have been a good option, due to the steep learning curve.

3. Applied process

3.1. Project management

Project management has mostly during this phase been focused on supervising that the work has advanced as planned. This has been done through short personal discussions, emailing, and group meetings, which during this phase so far only has needed about one hour more than estimated. The most difficult part has been to motivate all group members to really finish of their parts, and possibly this could have been even more successful.

3.2. Risk management

Risk management has mainly been done on a day to day basis, handling problems as needed. The most important risks are as defined in the last phase. *Reference: Risk Management document.*

3.3. Version management

Reference: Version management document

3.4. V-model testing

Reference: V-model testing document

3.5. Meeting practices

Reference: Communication and meeting practices document

3.6. Coding conventions

Reference: Coding conventions document

3.7. Integration and regular builds

Reference: Integration and regular builds document

3.8. Use-cases

Reference: Use Cases document

3.9. Customer and mentor relations

During this phase customer meetings could be limited to the beginning of the phase. After the final specifications with the customer during the first week all efforts were put into implementation of the remaining parts of the system.

The mentor has fulfilled his task to our full satisfaction, and has been willing to give helping advice always when needed.

4. Improvement

Based on the comments from the group members this phase has generally been felt as quite positive. And there has been a common feeling of that everyone has been working well on their parts. Taking into account that the next phase mostly focuses on delivery many of the improvements won't be that relevant for this project any more. Still some ideas how our work could be further improved:

- Still better communication
 - The need for communications has been constant throughout the project but the different kinds of work would have required different forums for the communication (e.g. during planning, implementation, bug fix and integration of different parts). This could be much further developed if such a project would be done in a company and the same people would continue to work together after the project.
- Better time management and respect for deadlines, hence better planned deadlines
 - The fact that some deadlines contained all too much work, the motivation to try to keep such a deadline decreased to almost none. This was a typical project management error, which was due to lack of experience.