Responsible for this document:
Emma Albertz
Linnéa Claesson

PUSS154219 v0.1 October 9, 2015

Team 2

Project Final Report

Authors of this document:
Emma Albertz
Linnéa Claesson
Jacob Mejvik

Contents

1 Reference Documents

Ref1 Veckoschema PUSS154251

Ref2 Gantt schema PUSS154252

2 Introduction

At the time this report is being written, the product is not completely finished. This document aims to analyse and evaluate the team's continuous work and the system to be delivered. Estimations on the last week's work and expected quality of the finished product have therefore been made.

Metrics data has been collected throughout the work, which is presented in section 3 and evaluated in section 4.1.

A questionnaire was sent out for all team members to fill out. The replies to this questionnaire is evaluated and presented in section 4.2. Suggestions for improvement based on these replies are then presented in section 5.

The team has during the work been divided into four subgroups with different responsibilities:

Project Managers Two people responsible for the entire project, planning and making sure everything is delivered on time and up to standard.

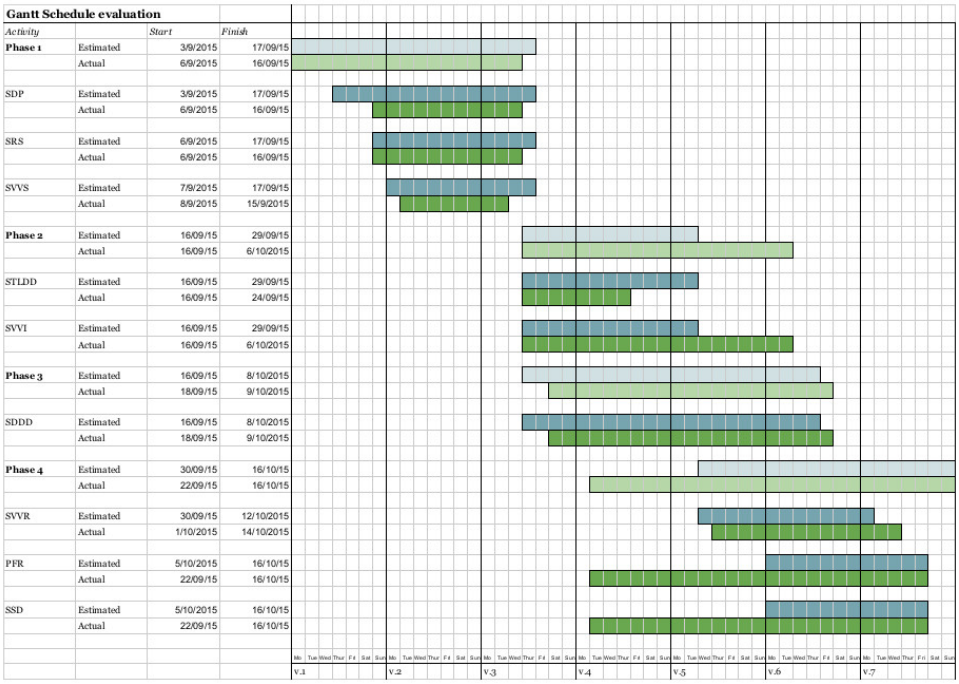
System Architects Responsible for the technical part of the project and communications between the developers and testers. Consisted of three people, one of whom was system manager.

Developers Eight developers have been responsible for the implementation of the application.

Test Group Six testers have been responsible for testing the developed system, one of whom has been acting as test manager.

The shared opinion within the team is that this has been a well executed project where everyone has contributed to their part to make sure that the delivered system is up to standards with the customer's request.

3 Project Metrics



Gantt Schedule evaluation			
Activity		*Start*	*Finish*
Phase 1	Estimated	3/9/2015	17/09/15
	Actual	6/9/2015	16/09/15
SDP	Estimated	3/9/2015	17/09/15
	Actual	6/9/2015	16/09/15
SRS	Estimated	6/9/2015	17/09/15
	Actual	6/9/2015	16/09/15
SVVS	Estimated	7/9/2015	17/09/15
	Actual	8/9/2015	15/9/2015
Phase 2	Estimated	16/09/15	29/09/15
	Actual	16/09/15	6/10/2015
STLDD	Estimated	16/09/15	29/09/15
	Actual	16/09/15	24/09/15
SVVI	Estimated	16/09/15	29/09/15
	Actual	16/09/15	6/10/2015
Phase 3	Estimated	16/09/15	8/10/2015
	Actual	18/09/15	9/10/2015
SDDD	Estimated	16/09/15	8/10/2015
	Actual	18/09/15	9/10/2015
Phase 4	Estimated	30/09/15	16/10/15
	Actual	22/09/15	16/10/15
SVVR	Estimated	30/09/15	12/10/2015
	Actual	1/10/2015	14/10/2015
PFR	Estimated	5/10/2015	16/10/15
	Actual	22/09/15	16/10/15
SSD	Estimated	5/10/2015	16/10/15
	Actual	22/09/15	16/10/15

Figure 1: Comparison between the estimated Gantt schedule and the outcome.

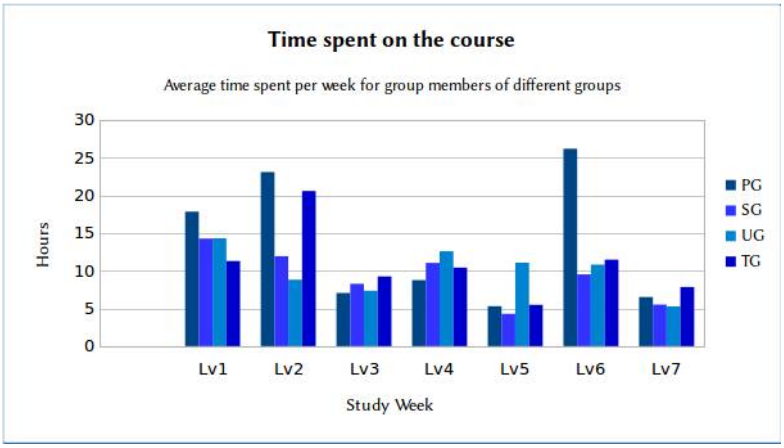


Figure 2: Average time spent per week fand person of each subgroup.

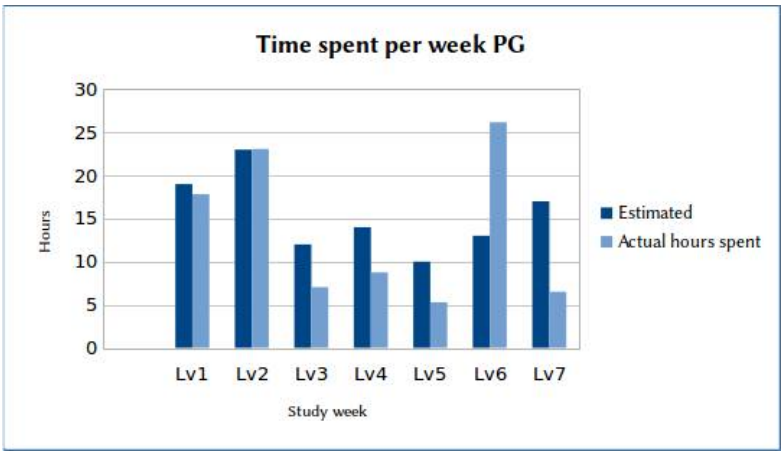


Figure 3: Average estimated and actual amount of hours spent by project managers per week.

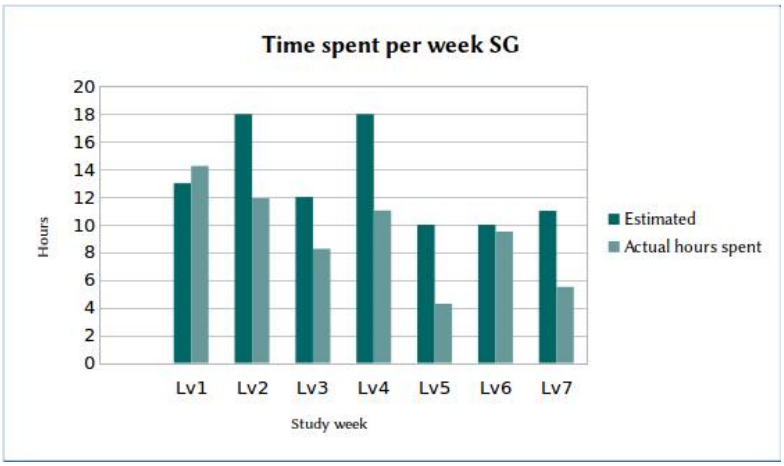


Figure 4: Average estimated and actual amount of hours spent by system architects per week.

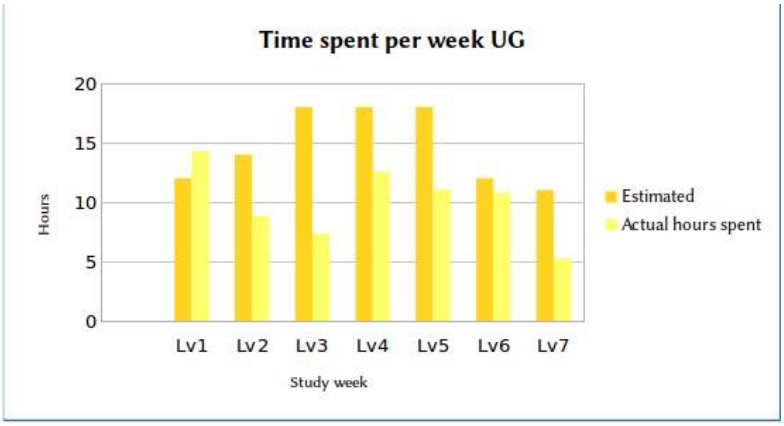


Figure 5: Average estimated and actual amount of hours spent by developers per week.

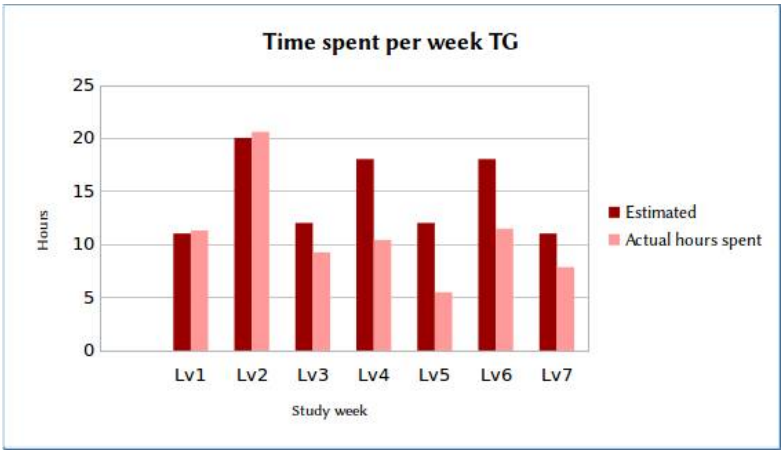


Figure 6: Average estimated and actual amount of hours spent by tester per week.

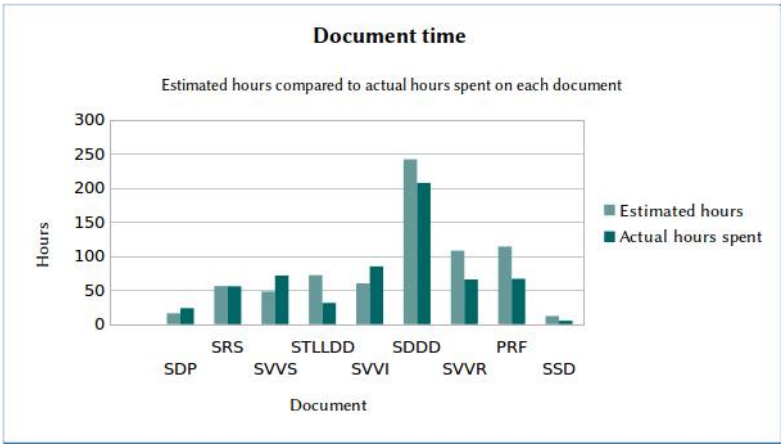


Figure 7: Estimated hours compared to actual hours spent on each document.

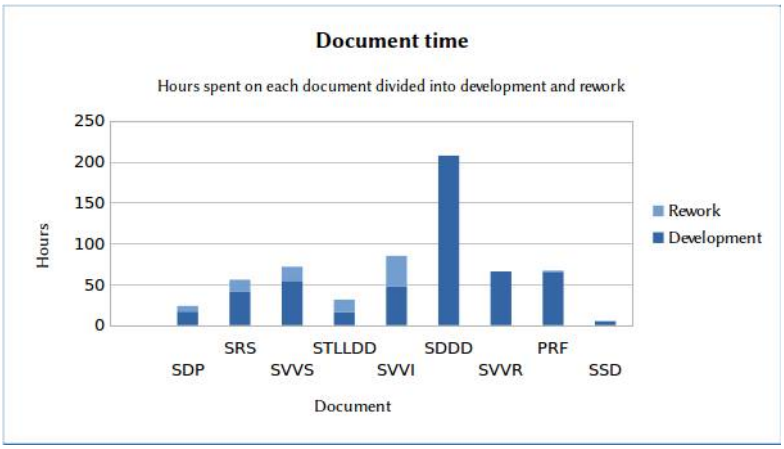


Figure 8: Hours spent on each document divided into development and rework.

Table 1: Table caption

Estimated			Result	
Activity	**h total**	**h/person**	**h in total**	**h/person**
SDP	16	8	24	12
SRS	56	8/4	56	13/2
SVVS	48	8	71	12
STLDD	72	8/6	31	10
SVVI	60	10	85	14
SDDD	242	25/14	207	26
SVVR	108	18	36	6
PRF	114	6	67	25/1
SSD	12	6	5	3

Om du vill referera till tabellen ovan skriv ”In table 1 bla bla bla”.

4 Project Evaluation

4.1 Evaluation of Project Metrics

Introtext

4.1.1 Ny sektion

4.1.2 Gantt Schedule

Overall, the dates set for the phases and the documents in the project has been followed with only smaller deviations. All according to the time plan made in the beginning of the project. However a few bigger deviations have occurred.

From the way it looks in the Gantt chart, SVVI took approximately a week longer than expected. This caused an extension of phase two. The reason for this was an administrative error. The document was actually ready within the planned time but was by mistake not put into baseline immediately.

Phase four started earlier than planned since project leaders found it appropriate to start preparing for the final reports as much as possible in advance. This to ease the workload in the final stage of the project as much as possible.

4.2 Evaluation of Delivered System and Performance of Team

As part of the project evaluation, a questionnaire was sent out for all team members to fill out. The replies to this questionnaire were used to evaluate the team’s performance during this project and are presented in this section.

Overall, the general consensus has been very positive and everyone seems very satisfied with both their own and the team’s contribution to the project.

4.2.1 Time Planning

The scheduling and time planning of the project have received very positive response from the team. The project managers put a lot of time into working out a reasonable schedule that would also make sure that the project was delivered prior to external commitments the team members

had at the end of the time period scheduled for this project. This lead to some tight deadlines, but the team was motivated to meet them due to it being to their personal advantage. At the time of this report being written, the deadlines have all been successfully met.

The Weekly Schedule and Gantt Chart, Ref1 and Ref2 respectively, produced by the project managers could have been referred to more often throughout the project. Overall though, the team members have stated that they have known what they needed to do and when they needed to do it. It has been stated that the pace of the project has been high, but that it has been good so the project can be finished before the external commitments previously mentioned.

Once a week a project meeting has been held where all team members have been expected to attend. The group meetings have been a great help in keeping the members of the team updated on upcoming deadlines and providing a chance for the project managers to make sure that the team is where it should be. Both the attendance and replies to the questionnaire have shown that these meetings were greatly appreciated and well carried out.

The developers did not have as much work in the beginning as the others, but this changed once they started implementing the code and they had to put in a couple of weekends to finish. This seems to have worked fine, even though there were some who could not attend these weekend group sessions.

The testers were sometimes dependent on the system architects to finish their reports before the testers could finish their own, which added a bit of pressure at the final stages of the writing of the reports. A slightly earlier deadline was often set for the system architects than the testers, but they were not given much extra time either since this would affect the system architects negatively.

4.2.2 Work Distribution

A responsible person for each task was decided at the group meetings. The division of the larger tasks into smaller has then been done internally within each subgroup, sometimes in collaboration with other subgroups (such as system architects and developers). This system has worked very well for this project. The project managers have always known who to talk to about each specific task to make sure everything is coming along as expected and that person can in turn make sure everyone within his or her subgroup is making the progress they should.

The responsibilities between the two project managers were easily shared. They had prior knowledges that complemented each other well and therefore found natural ways to divide the tasks between them. They regularly met to plan meetings and deadlines together and update each other on their respective tasks' progresses.

The same arrangements were made within the system architects group. Some had more technical skill sets and therefore took on greater responsibilities in those areas, whereas others had more experience in working with this kind of project were a lot of reports were to be produced at a certain standard and could therefore take on more responsibility in those areas.

Even though one of the system architects has acted as system manager, the organisation within their group has been very flat. They have met often to make sure everyone is on the same page with everything and they have all truly contributed to this project.

It was a bit unclear in the beginning how much responsibility the system architects should have. This could (and should) have been evaluated better and made clear from the beginning by the project managers, to avoid confusion. Overall though, they have known what needed to be done and no major issues arose due to this.

The developers have worked in pairs with their assignments. Most of them had no previous experience with Android development. Working in pairs really helped overcoming something that at first might have seemed very difficult, but still making each member responsible for a part

of the project. The division of what each pair should do came as a suggestion from the system architect group, who had prior experience with Android development and knew approximately how much time each task should take. There was one part of the project that was slightly larger than the others and the pair working on it had some difficulties, but the system architects stepped in and offered help so they could finish on time.

Overall the distribution of work among the developers has worked fine. Some of them have not seemed to prioritise this project and instead go and work on other things (everyone in the team has other commitments). This slowed down the development of the product, which could have been finished at an earlier stage had not some members postponed their tasks to the last minute. This was not a real problem though, considering the deadlines were met.

The work among the testers has also worked very well. There have been some inequalities in work load due to the fact only a couple of people within the group had previous knowledge of LaTex and GitHub. This meant that they had to pull a larger weight in fixing problems for the entire group.

4.2.3 Communication

One thing in particular that has worked very well throughout the project is the communication within the team, within each subgroup and also between the subgroups of the team. The developers have reported that there has been exceptional communications between them and the system architects, who have been a great help and contribution in the developers' progress and success.

A big contribution to the communication within the group functioning so well has been the weekly project group meetings. The meetings have been held at the same time and place each week, to avoid confusion and people missing meetings due to unclear scheduling.

At these meetings information to the team was given from the project managers and each subgroup has updated the rest of the team on their current status and progress. The questionnaire and the attendance showed that these meetings were appreciated and a great help in keeping track of what needed to be done and when. Who would be responsible for what part of the project, how issues should be handled and deadlines were set at these meetings.

The group meetings were also a place for discussion and gave an opportunity to ask the other team members questions and raise concerns. It also helped keep the project managers updated in how the work was going in each subgroup and making sure deadlines could be met.

Meeting protocols were constructed for each meeting and posted afterwards both on the team's joint Drive and Git repository, to be available for later reference and/or read by those who might have missed a meeting.

The subgroups have also had their own internal group meetings whenever they needed, to discuss topics not needed to be addressed at the larger group meetings. These topics include e.g. division of assignments and internal deadlines.

Information that could not wait to be brought up at the project meeting has been sent out by email. A Facebook group has also been used as a means for fast but secondary communications.

There was a discussion at the beginning of the project on whether or not to use Piazza as a means of communication within the group. Since it was not available from the start of the project other alternatives were used instead and by the time Piazza was working, these alternatives had already been established. There has been a difference of opinion within the group whether or not Piazza should have been used during the project and this should be evaluated for future projects.

Communications with the customer and experts have not been as well functioning as communications within the team. Especially the testers have reported that they could have avoided

some, in hindsight, simple issues by contacting the experts at an earlier stage of the project. The project managers and/or the system architects could also have had a better dialogue with the customer to continuously make sure that the system being developed was in fact the system ordered. The acceptance test has still not taken place at the time when this report is being written, but two formal reviews have been held with the customer prior to the acceptance test to make sure that the right system is being developed.

4.2.4 Experience and Knowledge

Both among the project managers and the system architects, the members had different sets of knowledge when coming into this project. Some had more technical skills, such as prior experience with the tools used (GitHub, LaTex and Android Studio) whereas others had more experience with this kind of administrative work and report writing. In both groups this worked very well, since they complemented each other and made the groups more knowledgeable as a whole than each individual. Everyone could make valuable contributions to the project.

Not many among the developers had previously worked with Android development, LaTex and/or GitHub. Since the external training that was provided came far too late in the process, the system architects had to step in and take a large part in training the developers in these areas. The external training also only covered Android development and not the other tools used for this project.

During the development of the product, the developers often chose to meet up and work together, even though the task was divided into smaller ones among them. This meant they could collaborate and help each other with problems. In addition to this, they could keep an open dialogue and everyone knew at what stage everyone else was at.

Som issues arose from the fact that not many among the developers had any previous experience with programming in a larger group. People would push code to git that could not be compiled, leading to problems for the rest of the group. A lot of time was wasted in dealing with these situations.

As previously mentioned, there was a large difference in prior working experience with the tools used (LaTex and GitHub) within the test group . This led to some doing a lot more work than the others when it came to solving problems generated by the tools they used, since they were the only ones who knew how to do it.

4.2.5 Technical Issues

Some technical issues were encountered during the work on this project, most of which were introduced outside of the group's control.

Bugs were discovered in the back end product, which the group had no control over or ability to fix and therefore had to work around.

Android Studio was not properly installed and available on the computers provided for this project, which meant that the developers had to use their own computers and install Android Studio on them. This caused issues for some members of the group who only owned stationary computers and therefore could not work together with the rest of the group. Some installation issues were also encountered but eventually fixed, but unnecessary time was spent on this process that otherwise could have been spent on development of the product.

The team was also supposed to use Piazza as a means of communications, both within the group and with the experts and customers. Piazza was not made available from the beginning of the project and therefore other tools were established before the team could start using Piazza and its advantages were then lost.

E-puss was also not available from the beginning, which caused some discomfort but no lasting consequences.

4.2.6 Documents and Final Product

When the questionnaire was sent out and this report written, the product was not yet completely finished. Estimations have therefore been made with respect to how satisfied the team will be with the final product. A lot of confidence was shown throughout the replies of the questionnaire, that the developers will be able to present an application that reflects the team's efforts and hard work.

The project managers are very satisfied with the quality of the documents so far delivered and believe without a doubt that with the help of the system architects and testers, the developers will be able to finish the application on schedule and to a satisfactory result.

The system architects have made sure that the Software Top Level Design Document (STLDD) has been followed. As always, things could have been implemented differently but at the time of the writing of this report, everything looks good and the system architects have faith in the developers ability to deliver the product ordered, based on the competence and hard work they already have put into it.

The system architects are also satisfied with their own work and results. They have put a lot of effort into this project and have made sure that the developers have had everything they needed throughout the development of the product.

The developers started to implement the code in parallel with the STLDD being written. This caused some issues when changes were made in STLDD that affected the code, but overall the product could be finished faster in doing so.

Most of the documents are based on templates provided by the customer, to make sure they follow a certain format and standard. This has been really helpful, since it has been clear from the beginning what the customer has expected.

There has been some dependence between some of the documents the team has delivered. It has been up to the system architects to manage this and make sure that everyone concerned has gotten the information they need to update their documents according to the changes made in another. A lot of work has been put into this and the other groups have seemed satisfied with the result. There were some miscommunication in the beginning of this process though, concerning documents produced by the system architects that affected documents produced by the testers. This was partly due to the project managers not communicating clearly from the beginning how these situations should be handled.

The developers are pleased with their contribution to the product. The main issues they have encountered in the development stage have been due to the back end and sensor not working properly. This has been out of their control, but a list of bugs in the back end has been delivered to the customer.

4.2.7 Problem Handling

The procedure for handling problems in the project has been very systematic. When a problem has been detected a problem report has been filed. Upon receiving a problem report the Change Control Board have investigated the problem, reviewed relevant documents and found a solution. Once a solution had been specified the responsibility to implement the changes was assigned and a deadline, usually within one day, specified. The process for handling problems has worked very well throughout the entire project.

The filed problem reports have generally been highly relevant, describing issues that significantly impacts the projects ability to deliver a high quality product with consistent documents.

For this reason, at the time of writing, all filed problem reports have been fully or partially accepted. In case of a partial acceptance the reason for deviation has usually been to keep a consistent and logic structure throughout all documents.

It should however be noted that the administrative load connected with handling problems has been high in comparison to the problem itself. This was anticipated, but in retrospect the load has been even higher than expected. The main reason for this is the rigorous procedure in connection with a desire to handle problem reports frequently and give quick feedback to relevant subgroups in order to shorten cycle times. At times this has meant that the Change Control Board has been forced to prioritize and had it not been for a well structured division of labour, they would have been hard pressed to deliver.

Furthermore, it has been noted that problem reports have had a tendency to occur more frequently towards the end of the project when the product nears completion. Problems that were injected later also tended to cause more widespread changes as more documents were set in baseline.

The problems that have occurred can mainly be divided into two categories, the first category being problems related to a mismatch between the STLDD and the SDDD. Some of these problem reports could have been avoided with a more thorough STLDD from the start. However, the changes have been of a minor character and the overall architecture has hardly been modified at all. With the benefit of hindsight, possibly the original design was somewhat overworked, containing a few unnecessary classes, in relation to the size of the application.

The second category is related to the back end. Changes had to be made due to the fact that the back end lacked basic functionality, such as being able to give accurate error messages, or that the functionality of the back end had been misunderstood in the early phases of development. Typically these errors propagated through several documents giving rise to various changes. Most of these issues were exceedingly hard to foresee and also resulted in multiple revisions in order to work around the constraints enforced by the back end.

In conclusion the problem handling has worked very well in terms of following the process, but has been quite labour intensive. Possibly some problem reports could have been avoided but the majority could not realistically be foreseen given the development model used.

5 Suggestions for Improvement

Even though the progression of this project has been a very good experience for all team members, there is always room for improvement and things that could have been done differently.

5.1 Development

One of the largest issues that came up was the lack of previous experience among the developers. The external training offered should be held at a much earlier stage than it was, which would have spared the system architects a lot of work. The developers could also themselves have taken responsibility in getting the knowledge and experience they knew they would need, especially since they did not have a lot of other assignments in the beginning of the project. The lack of knowledge was also known to the project managers and system architects. In hindsight it would have been a good idea if they had encouraged the developers more earlier in the project to retrieve the knowledge they would later need. They could have provided the developers with tutorials they could do to get the basics of Android development down and set a deadline for when this should be done. Instead, the system architects had to take on all this responsibility themselves in the beginning of the development phase.

The project managers feel that they could have had a more present role in the development stage of the project, e.g. attending some of the meet ups the developers had, and not just rely on the information received at the project meetings once a week.

5.2 Documents and Administration

The administrative part of the project was very large in comparison to the size of the developed system. A lot of time and effort was spent on producing the right documents and make sure that they kept a certain standard. This should be evaluated in the future to make sure that the time spent here is appropriate to the size of the ordered system.

To further increase the quality of the delivered documents, a more iterative process could have been used throughout the creation of them. This would add more time to the administrative part of the project though, which already was seemingly large.

The project managers could have more frequently referenced the time schedules, to make sure everyone knew about upcoming deadlines.

The project managers and/or the system architects should have had a more continuous contact with the customer, to make sure no misunderstandings would occur in the development of the system ordered. A weekly check up, perhaps via email, to ensure everything was on track as expected would be an easy and convenient way to achieve this.

The group managers of each group should have at an earlier stage established a good communication with the expert of their respective field, to avoid misunderstandings and simple mistakes.

A good idea would also be for the group managers to have a meeting once a week to update each other on each group's respective progress, in addition to the larger project meetings held once a week.

To avoid some people taking on a larger share of the work load, due to them having more knowledge and/or experience, the group manager should redistribute the tasks within the group. Problem solving is an equally important and demanding task as producing new material and lack of knowledge is not an excuse to put in less hours.

But as previously stated, all team members have done excellent work on this project and the project managers would be happy to take on any task together with them again in the future.