

# Project plan

Agnethe Soraa, Tomas Dohnalek, Jan Bednarik, Milos Jovac

Project adviser: Anh Nguyen Duc

August 30, 2013

## 1 Project customer

Netlight AS is a consulting company engaged in IT and management. They operates throughout Europe with offices in Stockholm, Oslo, London, Munich and Helsinki. The company was founded at 1999 and employs to 500 employees.

## 2 Project description

The customer wants a product to make the audience as a screen on a rock concert. We decided to name "digital lighter". The audience members at a rock concert should be able to download a simple application to their cell phone, and register this through a simple GUI. Behind the artist on stage there is a screen, with a simple camera on top. The camera is taking pictures of the audience. At special occasions the audience will be instructed, by the artist, to hold up their phones with the screen towards the stage. This is similar to holding up a lighter like people did in the old days to create a special atmosphere at the concert. We want to digitalize this by giving the audience a chance to use their cell phones instead of lighters.

On control a signal the application will fill the entire mobile screen with a single color. The control signal can as an example say: "all pixels white". The signal will be specific for each application. Each mobile will be a pixel in a larger picture, which will be presented on the big screen. What kind of picture the audience can create will depend on the number of people in the audience.

As a motivation for the audience to hold up their phone, the camera on top of the screen will take pictures of the audience. In that way the audience can see a reflection of them selves, and see what kind of picture they are creating together.

### **3 Required work**

- ?

### **4 Project scope**

After third meeting with our customer we decided to lay out the scope of the project that is tangible and doable under 13 weeks. Customer idea although good and innovative has one flaw - it is a huge undertaking! As we operate under certain limitations we explained in General Terms section, subsection Limitations, we agreed with customer on next terms: - Take project title as domain of work and not final product, - Scale down problem but attack all main problems of the domain, - Disregard scaling of product, - Disregard some of problems that are not important for final prototype, but talk with a customer before make this kind of decision.

### **5 Project architecture**

Product can be logically divided into two sections – client and server application. Client side should be used by the audience whereas server side should be used by the concert manager. Both kind of applications will communicate through network, there is no need of internet connection.

Each application can be divided into more detailed architecture; we will focus on that in next sprint.

### **6 Measurement of Project Effects**

To measure success of our end-product we have to set up some criteria to be fulfilled. The product should pass all test-cases and function according to customer's requirements.

### **7 Planned workload**

Compendium proposed week workload 25 person-hours per week. During our internal meeting we have decided that each member will spend 30 hours per week because our team consists only of 4 members. We agreed on fixed daily working hours so that we could distribute the workload through the whole semester. We will do daily stand-ups according to Scrum methodology.

## 8 General Terms

### 8.1 Methodology

After fast research and consultation with our mentor and customer(He have experience with CDP student group from last year), we decided to use SCRUM methodology for a few reasons. Requirements and scope of the undertaking were not that precisely defined at the time we had to decide on the methodology. Our approach therefore could not be founded on the sequential methodology as is "waterfall". We wanted to have frequent meetings with our customer and include him in development process. Sprints in SCRUM allow us just that - to have meeting with customer at the end of every sprint, and plan next one together. Sprints doesn't have to be the same length so we can better manage developing process and risks, and we will be able to have as much sprints as possible under time limit of 13 weeks. SCRUM will give us derivatives that we can enhance in increments and allow us to gradually reduce the risk and keep our customer informed about our progress. SCRUM is also a very popular approach in the software industry, so it is a good choice to learn it.

### 8.2 Tools selections

For Scrum support and issue tracking we use Gravity Tool<sup>1</sup>. The tool is right now in Beta but it is free to use and have all features we needed from proposed AgileZen. CHANGE(30.08.2013.) As Gravity missing some of the features we didn't know we need it was hard to procide in it. New tool we established is Target Process 3(<http://www.targetprocess.com/3/>). Tool is also free and is not in Beta.

For collaboration on Minutes, Project Plan and other documents we use GitHub. This tool was proposed by out customer and it is popular free collaboration tool. For document editing we agreed on LaTeX. For group resources and links we use Facebook groups and for managing schedule we use Google Calendar.

### 8.3 Limitations

We are developing this project under a few technical, resource, time and knowledge limitations.

Our biggest limitation is the image processing part. Half of the team has no experience with this, and the other half has little experience. Their experience is mostly theoretical information about the subject, and practical experience is preferred. We are aware of this limitation, and our plan is to learn by doing. We are going to start developing, and teach ourself while coding. We chose this approach because we do not want to spend more time than necessary doing research.

---

<sup>1</sup>[www.gravitydev.com](http://www.gravitydev.com)

Another limitation is lack of experience with Mobile development within the development team. All of the team members have Android phones, and to be able to test our application, we have to develop an Android application. Only one team member have experience with this.

If we are not scaling down the project, then we do not have all necessary resources to test the system. As an example we do not have a huge audience. Also we do not have access to a big screen etc. This is also a limitation.

As this course last for a 13 weeks, it is normal that we have to make some trade-offs. This project is technically difficult, and there is a limited amount of time.

## 9 Schedule

general paragraph about sprint length

### 9.1 Phases

**Sprint 0 (ends 6th of September)**

**Sprint 1 (ends 20th of September)**

**Sprint 2 (ends 4th of October)**

**Sprint 3 (ends 18th of October)**

**Sprint 4 (ends 1st of November)**

**Sprint 5 (ends 15th of November)**

### 9.2 Gantt chart

### 9.3 Milestones

We set milestones in order to more accurately ascertain whether or not the project is on schedule. After finishing some group of the milestones we will have running prototypes. There are 8 prototypes In total we agreed on. Prototype is a milestone by itself. Last or 8th prototype implies that all other milestones are reached. Although the order of milestones may look like a waterfall approach that is not the case. Prototypes are the milestones as we said before but they are also a logical and operational units and therefore some logical order of the milestones had to be ordered. Furthermore, milestones are not necessary connected to sprint, and group of milestones that prototypes are "composed" of do not have to be finished in specific order.

- **Sprint 0**

- Project Plan
- Sprint 1
- Architecture design
- Simple GUI client & server
- Sprint 2
- Implementation of client
- Set up protocol for client & server
- Sprint 3
- Finish GUI client & server
- Sprint 4
- Implementation of server
- Image processing
- Synchronization
- Sprint 5
- Evaluation
- Sprint 6
- Project Report
- Demo
- Presentation

## 10 Organization

### 10.1 Role Assignment

To assign roles according to our skills and previous experience we have decided to make survey of relevant knowledge. Results of this survey can be seen in table 1. This table was used as a base for our role assignment, but in practice the only roles used are communications and the scrum master. We decided to embrace this as a development team were we are all equal. The assigned roles can be seen in table 2. In the beginning we have decided to share other roles proposed by compendium.

Figure 1: Gantt Chart: Allocation of sprints into weeks.

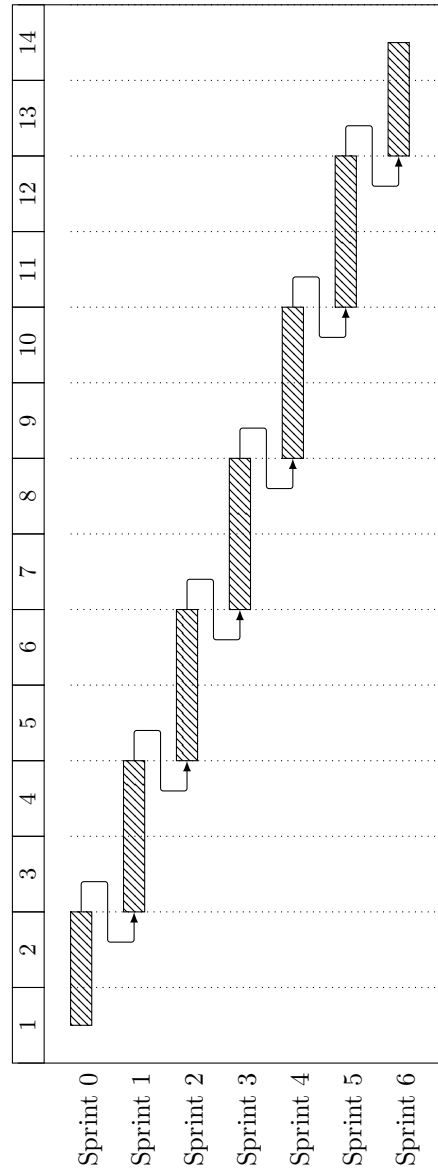


Table 1: Skills and previous experience table. Coding: ● expert, ● experienced, ● neutral, ● little experience, ● no experience

	Agnethe	Tomas	Milos	Jan
<b>Leadership</b>	●	●	●	●
<b>Scrum</b>	●	●	●	●
<b>Mobile software development</b>	●	●	●	●
<b>LaTeX</b>	●	●	●	●
<b>Network programming</b>	●	●	●	●
<b>Image processing</b>	●	●	●	●
<b>Java</b>	●	●	●	●
<b>C++</b>	●	●	●	●
<b>Testing</b>	●	●	●	●

Table 2: Assigned roles and their responsibilities

Role	Person	Responsibility
<b>Project Leader</b>	Milos	Responsible for progress of the project according to the plan. Distributes work to group members. Has final call in arguments.
<b>System Architect</b>	Tomas	Check consistency and analyze all layers of the product.
<b>Scrum Master</b>	Agnethe	Leads the scrum stand-ups.
<b>Communication Responsible</b>	Jan	Responsible for communicating with customer and supervisor. Regularly send meeting minutes, agenda and other documents to customer and supervisor.
<b>QA Responsible</b>	Tomas	Ensure a quality of all documents and end-product.

## 11 Risk management

In the table below the consequence and possibility in a number between 1-10. The risk, or the riskfactor, is the consequence multiplied with the possibility. The risks in this table is very obvious ones.

Other risks we can see from analyzing the skill table. If for example the only two persons on the team who are familiar with image processing is away, then this will be a risk. There is a great possibility that this kind of task will take longer time then planned for.

Also if some of the rows in the skill table all were painted red, which would imply that the team had experience with this, then this would be a risk. In such a case, we would have to talk to our customer and consider scaling down this task.

## 12 Quality Assurance

Specific guidelines and practices were adopted so that the team and the customer could continuously verify that the project development keeps the right direction and that the requirements are being fulfilled.

### 12.1 Customer collaboration

Since the scope of the project and the actual requirements are determined by the customer it is essential to establish tight collaboration practises and even involve the customer in the collaboration tools the team uses.

**Meetings** Considering this guidance we agreed on the weekly meetings. As the customer is located in Oslo the meetings will be carried out using the video-conference devices provided by the NTNU (accessible in the Accenture Lab) and Skype software. After each

**Email** Standard email messages will also serve the purpose of the communication mean, both team and the customer agreed on responding to the queries as soon as possible.

**Documents** - minutes - access to Drive

**Other tools** - Gravity, Skype conference, Google calendar

- meetings - one person communicates - customer decides about stories, accepts/rejects deliverable - quality is determined by customer who has the opportunity to evaluate the product (in its current state) after every sprint.

### 12.2 Supervisor collaboration

- one person communicates



Table 3: Handling risks

Event	Consequence	Possibility	Risk	Reactive Measures	Proactive Measures
Someone gets sick	4	5	20	Other people do more work.	Free weekends
Coding problems	4	7	28	Talk to supervisor & Guru office	preparing for the task
Testing problems	4	4	16	Talk to customer about reformulating requirements	Double check requirements with customer
Implementing things we are not supposed to	7	6	42	Try to adopt functionality or start all over	Don't do anything that is not in backlog and keep good communication with customer
Dead end with technologies	8	8	64	Talk to supervisor & Guru office	Do thoroughly research
Unrealistic time estimate	7	8	56	Work overtime	Planing poker
Frequent changes in requirements specification	6	3	18	Renegotiate with a customer	Try no to change finished modules and keep weekly meetings with the customer
Customer too ambitious	9	5	45	Renegotiate with a customer	Keep customer informed about what to expect
Hardware problems	9	3	27	Obtain a new one	Keep your devices updated

### **12.3 Team collaboration**

- continuous doc writing - daily standups