Technische Universiteit Delft

TI3800 Bachelorproject

A non-centralized approach to Video on Demand on mobile devices

Plan of Action

Authors:
Martijn Breet
(1265458)
Jaap van Touw
(1380753)

Supervisor:
Cor-Paul Bezemer

July 31, 2013

Contents

Preface

This document forms the Plan of Action which is part of the Bachelor Thesis that will be performed by Jaap van Touw and Martijn Breet. The Plan of Action will give insight on which problem will be solved during the project. The responsibilities of the project members as well as the client will be covered. Additionally insight will be given into the overall process and phasing of the project.

Chapter 1

Introduction

1.1 Plan of Action

This document describes the goal of the proposed Bachelor project and elaborates on how this goal will be accomplished, as well as the timeframe in which it needs to be completed. The document is structured as follows. The remaining part of this chapter will focus on preliminary background information that will explain the context in which the proposed Bachelor project will be performed. This will include a brief description of the research department and a background information on the topic of the project, as well as the causes that led to the project assignment. Chapter 2 will give a detailed description of the project. The project stakeholders are identified, as well as the main research question that the project will focus on answering. Additionally, a list of all the deliverables for the project will be covered. The last part of the chapter will cover the general requirements and constraints of the project, as well as the conditions under which the project will be performed. Chapter 3 will focus on the approach that will be used to answer the research question and will elucidate the project planning. The administrational part of the project will be covered in Chapter 4. Finally, Chapter 5 will focus on the measures that will be taken to assure the quality of the project.

1.2 Department description

The Parallel and Distributed Systems group (hereafter: PDS) is a research group within the Software and Computer Technology (hereafter: SCT) department, which is part of the Faculty Electrical Engineering, Mathematics and Computer Science (hereafter: EEMCS) of Delft University of Technology. The research within the PDS group concentrates on the modeling, the design, the implementation, and the analysis of parallel and distributed systems and algorithms. Most of this research is experimental: the aim is to build prototypes of systems, preferably used in the real world, to demonstrate the quality of the proposed solutions. The main research areas of the PDS group are Peer-to-Peer (hereafter: P2P) systems and online social networks, massive multiplayer online games, grids, clouds, multicore architectures and parallel programming.

1.3 Background and cause of assignment

Distribution of radio and television programs, movies, music, ringtones, games, and various data applications to the general public is possible today via a variety of dedicated networks and special end-user terminals. As broadband Internet becomes ubiquitous in both desktop and mobile device environments, all content distribution services will be combined and conveyed to the general public via a common pipeline, the Internet. Today several technologies are used for the media distribution across the Internet: unicast, IP multicast, content distribution networks, and most recently Peer-to-Peer. P2P is considered by many as an efficient, reliable, and low cost mechanism for distributing any media file or live stream, and it is used extensively[1]. Much of the current research activities in P2P within the PDS group are centered around Tribler[2]. Tribler is an application that enables its users to find, enjoy and share content through a P2P network. Tribler builds on BitTorrent[3] and is available on desktop environments. Currently mobile internet traffic continues to consistently gain on desktop traffic in terms of volume[4] and mobile traffic is estimated to surpass traffic from wired devices in 2017[5]. In response to this growth and to meet the increasing demands of the market, the development of a mobile version of Tribler would be of great value.

[1] http://www.ipoque.com/sites/default/files/mediafiles/documents/internet-study-2008-2009.pdf
[2] http://www.tribler.org
[3] http://www.bittorrent.com
[4] http://gs.statcounter.com/mobile_vs_desktopwwmonthly200812201306
[5] http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.pdf

Chapter 2

Project Assignment

2.1 Introduction

In this chapter, key aspects of the project are set out such as the main research question that the project will be focused on answering, the goal of the project, as well as an exact formulation of the project assignment. General requirements and constraints for the project are also given here.

2.2 Client

The client is Dr. Ir. Johan Pouwelse, he is Assistant Professor at the Parallel and Distributed Systems Group of the Faculty of EEMCS, Delft University of Technology, and is also co-founder of Tribler. Moreover, he is Scientific director of several P2P research initiatives with a total budget of 26 Million Euro.

2.3 Stakeholders

The Client:

Dr. Ir. Johan Pouwelse
J.A.Pouwelse@tudelft.nl
+31 (0)15 27 82539
Room: HB 07.290
Mekelweg 4
2628 CD Delft

Project Supervisor:

MSc. Cor-Paul Bezemer
C.Bezemer@tudelft.nl
+31 (0)15 27 82467
Mekelweg 4
2628 CD Delft

Bachelor Project Coordinator:

Martha A. Larson
M.A.Larson@tudelft.nl
+31 (0)15 27 87357
Room: HB 11.040
Mekelweg 4
2628 CD Delft

2.4 The Research question

The project will be focused on answering the following main research question:

"How can we make video-on-demand available for mobile devices using a non-centralized approach?"

2.5 Goal

The goal of the project will be to create a prototype application for mobile devices, that allows users to enjoy video-on-demand through the Internet, using a non-centralized network architecture. The application will allow users to search for a video, which will start playing after the user presses a play button. Additionaly, seeking functionality will be provided.

2.6 Assignment Formulation

In this project, a prototype of the P2P-based video-on-demand mobile application will be realised, that will allow users to search for videos on the Internet. Once the user has found a video to his or her liking, and issued a play command, the application will play the video by means of streaming it through a P2P-based network. The application will also feature the possibility to seek to different parts of a video, by means of a slider, that can be adjusted by the user.

2.7 Deliverables

During and after the completion of the project, the following products will be delivered:

1. Orientation Report.
2. Requirements Analysis Document
3. Architectural Design Document
4. Technical Design Document

5. Test- and Implementation Plan
6. Implementation
7. Source code evaluation by SIG[1]
8. Final report

The deliverables above are further defined in section 3.3.

2.8 Requirements and constraints

The final product will offer the features that are described in section 2.6 and should be considered a "prototype". The prototype will offer video-on-demand, including built-in search functionality. The development of the prototype will be targeted to the Android platform.

The exact functional- and nonfunctional requirements, as well as the constraints will be specified in the Requirements Analysis Document.

2.9 Conditions

The project members listed under section 4.2 will deliver the products listed in section 2.7 within the period starting from the 17th of July until the 2nd of October 2013.

The client will facilitate the project members by supplying them with all the resources needed for the development of the deliverables, such as (mobile) testing devices and required office space.

The project members will have weekly meetings with the supervisor (see section 2.3) to discuss the status and progression of the project, as well as to receive feedback on completed work. These weekly meetings are also part of the development methodology which will be elucidated in section 3.6.

[1] http://www.sig.eu

Chapter 3

Approach

3.1 Introduction

In this chapter insight is given into the approach that will be used to complete the project, including a brief description for every project phase and its deliverables. This chapter will also focus on the methodology that will be used during the project. Finally, the project planning will be elucidated.

3.2 Orientation Phase

In this phase, the research question is further crystallized into subquestions, which will create a better understanding of the problem as well its scope. After this is clarified, research will be conducted, by exploring existing (partial) solutions that can contribute towards a general solution for the problem. The findings of this research will be stated in the Orientation Report.

3.3 Design Phase

As a first step in the design phase, functional-, nonfunctional requirements and constraints for the prototype will be elicited. From these requirements, usecases will be derived that convey how the system should interact with the user. Based on these requirements and use cases, an architectural design document will be created, consisting of a description of the proposed software architecture including subsystem decomposition, persistent data management, global resource handling, concurrency, software control and boundary conditions. Next, a detailed description of the packages, class diagram and a specification of the classes and methods is given in the Technical Design Document. Finally, a Test- and Implementation Plan is created that describes how the different features of the prototype will be tested and implemented.

3.4 Implementation Phase

Building the application is the main part of the project and consists of implementing the different components that were derived in the Design Phase. The

source code of the prototype will be sent to the SIG for a thorough review on the 5th of September. The feedback provided by the SIG will be used to improve the code during the implementation phase.

3.5 Release Phase

In this final stage of the project, all the documents that were created during the previous phases of the project will be bundled into one final report. This will include a general conclusion and evaluation. After handing in the Final Report to the stakeholders, the source code of the prototype is sent to the SIG one more time for a final evaluation. The finished prototype will then be presented to the client, bachelor coordinator and supervisor.

3.6 Methodology

3.6.1 Scrum

During the project several methods will be put into practice. One of these methods, namely Scrum[1], will be used in all phases starting from the design phase. Scrum is an iterative and incremental Agile method. Scrum uses sprints in which the team goes through the process of adding functionality to the software, always maintaining a working version of the prototype. Given the relatively small timespan of the project, sprints of one week are chosen to ensure progress is measured frequently. Traditionally, the Scrum method defines a number of roles assigning different responsibilities to each of the team members. Since the team for this project solely consists of two persons, no specific roles are assigned. At the start of each sprint, a meeting; called a sprint planning, will be held. These meetings will be attended by the team members, as well as the supervisor. In this manner, the supervisor gets a better insight into what progress is made and is able to provide more meaningful feedback on the previous and upcoming tasks. In the sprint planning the following is discussed:

- Which tasks have been completed during the last sprint.
- Encountered impediments, if any.
- Decide on which tasks have to be done in the upcoming sprint.
- Determine the time it will take to complete the tasks and assign these to the team members.

The daily scrum meetings, also known as 'standups', will only be attended by the team itself. During these meetings, the team will briefly discuss what each person did on the day before, what each person is going to do and if there are any impediments that need to be overcome. Furthermore, bi-weekly demo sessions wil be held with the client, to keep the client informed on the progression that is made.

[1] http://www.scrum.org

3.6.2 MoSCoW

MoSCoW is the de facto standard in prioritizing a list of requirements into the following categories:

- Must have: requirements that must be satisfied in the final solution for the solution to be considered a success.
- Should have: high-priority requirements that should be included in the solution if possible.
- Could have: requirements which are considered desirable but not necessary.
- Would have: requirements that will not be implemented in a given release, but may be considered for the future.

The MoSCoW method will be used to prioritize the elicited requirements in the Requirements Analysis Document.

3.6.3 Test Driven Development

Conventionally, testing used to be done after the implementation of the software, by means of writing and executing test cases. This approach however, is criticised because often the tests are written based on the source code, instead of the functional requirements. A passing test in that case does not guarantee whether or not the function under test satisfies the functional requirements. Designing and writing the test cases beforehand ensures that the functional requirements are validated in an early stage. This approach is called Test Driven Development (hereafter: TDD). Another advantage of TDD is that it works well in combination with Agile methods such as the previously described Scrum method, because each time a new function is implemented, the associated tests can be immediately executed. As a consequence, the system that is being developed can be in a validated and verified state at all times during the development process.

3.7 Planning

A Gantt chart of the project planning can be found in Appendix A. The chart is complemented with a timeline, for a quick overview.

Chapter 4

Project Design

4.1 Introduction

This chapter will cover the administrative aspects of the project such as the project members, reporting, financing and facilities.

4.2 Project members

The project members that will work on the project are Martijn Breet and Jaap van Touw. Both members are required to work at least 40 hours per week on the project. In addition to that, both members are required to make an equivalent contribution during every phase of the project, in order for both members to gain the same amount of experience with all types of activity (requirements analysis, design, implementation, etc.). A short introduction to each of the project members is given below, in which the previous and current activites of each member is briefly described, together with their current contact information.

Jaap van Touw
Jaap is a Computer Science bachelor student at the faculty of EEMCS. His main programming language is Java and before starting this project he worked at the DUT Racing Team as software engineer.

J.vanTouw@tudelft.nl
+31 (0)6 505 37 951
Room: HB 07.240
Mekelweg 4
2628 CD Delft

Martijn Breet
Martijn is a Computer Science bachelor student at the faculty of EEMCS and has over 4 years of working experience in the field of software engineering. Before starting this project, he spent one year at the Delta Lloyd Solar Boat Team as full-time board member and software engineer.

M.S.Breet@student.tudelft.nl
+31 (0)6 158 86 822
Room: HB 07.240
Mekelweg 4
2628 CD Delft

4.3 Financing

All work within the project will be done on a voluntary basis. No specific budget is available for the project, and all the necessary work has to be be conducted using existing resources (see section 4.5).

4.4 Project Reporting

In order to keep the client up to date, both oral and written communication will be utilized. Informal meetings with the client will be held in person on a weekly basis. Written communication will be performed by means of using an IRC[1] channel , as well as per e-mail. Apart from the client, communication with the supervisor will also be done orally and in writing. Weekly sprint meetings will be held with the supervisor. The agendas and minutes of these meetings will be documented. All project documentation will be written using the LaTeX[2] typesetting system, and will be released as LaTeX source code, as well as in PDF[3] format. All released project material, including source code and documentation will be uploaded to an online GitHub[4] repository which is publicly available at all times.

4.5 Resources

The project members each already own a laptop, which they will use for all the required software development and document creation. The client has arranged an office room at the department of the PDS group where the project members can work full-time. All the software that is required to perform the project tasks is open-source and freely distributed over the internet. In case additional hard- or software is required during the project, the project members will ask the client for the required funding of these procurements.

[1] http://www.irchelp.org/
[2] http://www.latex-project.org/
[3] http://www.adobe.com/products/acrobat/adobepdf.html
[4] https://github.com/javto/bsc-project

Chapter 5

Quality Assurance

5.1 Introduction

This chapter will focus on how different techniques and methods will be used to assure the quality of the final product and help improve the Bachelor project itself.

5.2 Quality

5.2.1 Documentation

During the overall process, several documents (see section 2.7) will be created in which design choices are made and justified. These documents will be reviewed by both the supervisor and the client, after which the feedback will be processed by the project team.

5.2.2 Version Control

An existing version control system will be used during all phases of the project to keep track of the history of every created document and source code. One of the benifits of using a version control system is that in case an undesirable change is made to a document, one can revert back to any previous version of the document. The system of choice for this project will be Git[1], a widely used open-source distributed version control system. It is a system the team is already proficient with and also what the Tribler-team is using at the moment.

5.2.3 Code Review

During the implementation phase, the source code that is written by each project member will be briefly reviewed by the other member. This way, feedback can be obtained that can help improve the overall quality of the source code.
A complete source code review will be done by the SIG (see section 3.4), in which the quality of the software is professionally evaluated. By processing their

[1] http://git-scm.com/

feedback and recommendations, the project members can improve the quality of the software.

5.2.4 Software Testing

During the implementation phase we will use TDD (see section 3.6.3) to ensure that the software is validated and verified at any given point in time.

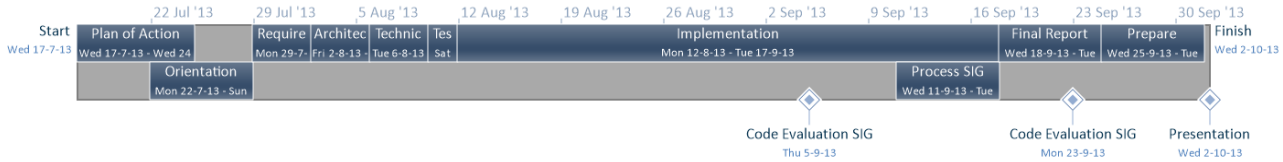
5.2.5 Evaluation

After the project, an evaluation will be written by the project team to reflect on how the process went. In this reflection, focus is put on how the overal process, as well as the approach of each teammember can be improved. Additional feedback on the bachelor project itself will also be provided. The evaluation does not necessarily increase the quality of the final product itself, but it does increase the quality of future work of the project members, the bachelor project and the final product in the long term by showing what more can be done to expand or improve the final product.

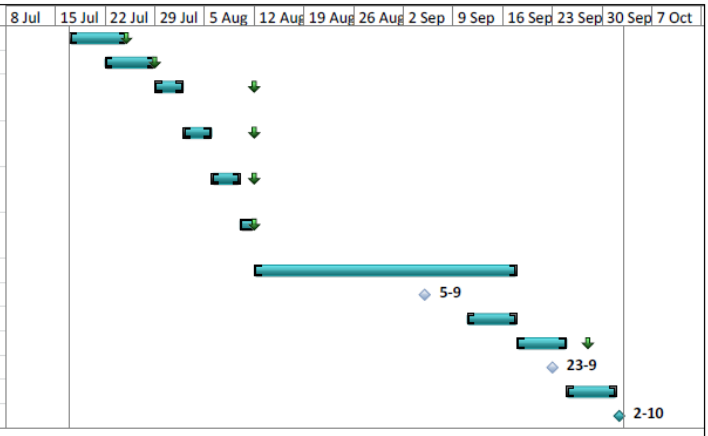
Appendices

Appendix A

Project Planning



ID	Task Name	Duration	Start	Finish
1	Plan of Action	1,14 wks	Wed 17-7-13	Wed 24-7-13
2	Orientation Report	1 wk	Mon 22-7-13	Sun 28-7-13
3	Requirements Analysis Document	4 days	Mon 29-7-13	Thu 1-8-13
4	Architectural Design Document	4 days	Fri 2-8-13	Mon 5-8-13
5	Technical Design Document	4 days	Tue 6-8-13	Fri 9-8-13
6	Test- and Implementation Plan	2 days	Sat 10-8-13	Sun 11-8-13
7	Implementation	5,29 wks	Mon 12-8-13	Tue 17-9-13
8	Code Evaluation SIG		Thu 5-9-13	
9	Process SIG Feedback	1 wk	Wed 11-9-13	Tue 17-9-13
10	Final Report	1 wk	Wed 18-9-13	Tue 24-9-13
11	Code Evaluation SIG		Mon 23-9-13	
12	Prepare presentation	1 wk	Wed 25-9-13	Tue 1-10-13
13	Presentation	2 hrs	Wed 2-10-13	Wed 2-10-13



Project: Mobile VoD

Task Start Deadline

Milestone Finish