



# **StudyAbroad Project Plan**

**Version 1.0**

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

## Revision History

<b>Date</b>	<b>Version</b>	<b>Description</b>	<b>Author</b>
2012-10-24	0.01	Initial draft	DSD staff
2012-11-02	1.00	Project plan v1.0	StudyAbroad project team

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

## Table of Contents

1.	Introduction	5
1.1	Purpose of this document	5
1.2	Intended Audience	5
1.3	Scope	5
1.4	Definitions and acronyms	5
1.4.1	Definitions	5
1.4.2	Acronyms and abbreviations	6
1.5	References	6
2.	Background and Objectives	6
2.1	Project goals	6
2.2	Project requirements	6
2.2.1	Data sources	6
2.2.2	Visual presentation	6
2.2.3	Preferences system	7
2.2.4	Recommendation system	7
2.3	Project milestones	7
2.4	Project deliverables	7
2.5	Project testing	8
2.6	Product delivery	8
3.	Organization	8
3.1	Project management	8
3.2	Project group	9
3.3	Project responsibilities	9
3.3.1	Project leader (manager)	9
3.3.2	Team leader (manager)	9
3.3.3	Documentation manager	9
3.3.4	Risk manager	9
3.3.5	Backup manager	9
3.3.6	User interface manager	10
3.3.7	Requirements manager	10
3.3.8	Lead developer	10
3.3.9	Testing manager	10
3.3.10	Database manager	10
3.3.11	System architect	10
3.3.12	Virtual machine manager	10
3.3.13	SVN manager	10
3.3.14	Data sources manager	10
3.4	Project tools	11
3.5	Customer	11
3.6	Supervisor	11
3.7	Others	11
4.	Development process	11
4.1	Overview – Personal RUP	11
	Detailed process	12
4.1.1	Inception phase	12
4.1.2	Elaboration phase	12
4.1.3	Construction phase	12
4.1.4	Transition phase	12
4.2	Roles	12

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

4.3	Progress tracking	13
4.4	Quality assurance	13
5.	Deliverables	13
5.1.1	Remarks	15
6.	Inputs	15
7.	Project risks	16
8.	Communication	17
8.1	Synchronous communication	17
8.2	Asynchronous communication	17
9.	Configuration management	17
10.	Project plan	18
10.1	Time schedule	18
10.1.1	Remarks	18
10.2	Activity plan	20
10.2.1	Remarks	21

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

## 1. Introduction

StudyAbroad project aims to develop a system that will help students in choosing a university and the city in which they want to study. The project goal is to collect information from different data sources on the Internet and use it to develop a recommendation system which will guide the user towards a decision.

### 1.1 Purpose of this document

This document is intended for members of the team involved in developing StudyAbroad application. The document describes organization of the team and future plans for the project. It was written at the beginning of the project and will be updated in the future if there will be need for it. Revision of the document should be done if there are changes in team organization or in project plan. These plans are not strictly defined and there are possible changes in future revisions. Since this is a basic document, all documents in future should be based on the information contained in it. Any differences in the information that the future documents will contain should be corrected in future revisions.

### 1.2 Intended Audience

This document defines the roles of the team members and helps them to define their tasks. It is very important in the initial phase of the project, but also plays an important role in future phases because it defines guidelines and methodologies that will be used in the project from now on. All members of the development team including the supervisor use this document as the basis of the project.

### 1.3 Scope

This document provides an overview of the information essential for the organization and work on the project. The document describes project roles for each member of the team, time schedule, deadlines and risks. There are also included tools used for communication and collaboration in the project, and the details of the technology that will be used to develop the system.

### 1.4 Definitions and acronyms

#### 1.4.1 Definitions

<b>Keyword</b>	<b>Definitions</b>
StudyAbroad	The project name
Planbox	Agile project management tool that allows members of multiple business functions to plan, collaborate and deliver projects.
MS Project	Project management software program, developed by Microsoft, which is designed to assist a project manager in developing a plan.
Subversion	Software versioning and revision control system
Alpha prototype	First iteration prototype
Beta prototype	Second iteration prototype
Acceptance test	Acceptance test is a test conducted to determine if the requirements of a specification or contract are met.

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

#### 1.4.2 Acronyms and abbreviations

Acronym or abbreviation	Definitions
FER	Faculty of Electrical Engineering and Computing
MDH	Mälardalen University
POLIMI	Politecnico di Milano
SVN	Subversion
DSD	Distributed software development
RUP	Rational Unified Process
VM	Virtual machine
GUI	Graphical user interface
ER Diagram	Entity–relationship diagram

### 1.5 References

Project homepage: <http://www.fer.unizg.hr/rasip/dsd/projects/studyabroad>

## 2. Background and Objectives

### 2.1 Project goals

The goal of the project is to build a web application which will provide a service to the user in a form of a recommendation system for deciding where to go to study abroad. The application will gather data needed to make that decision from various open data sources and present the data to the user in a organized and visually attractive user interface. The application will allow the user to input data on a number of options so the system can provide the user with choices which are suitable to his preferences.

Additionally an important project goal is to learn how to overcome obstacles in the process of developing software in a distributed environment.

### 2.2 Project requirements

In general there are several main requirement categories of our project. There is the collection of data that needs to be gathered from various data sources, presentation of that data, the filtering system for narrowing down the search and the recommendation system.

#### 2.2.1 Data sources

Several open data sources will be used for providing information in different categories like:

- General information
- Financial information
- Publications
- Maps and Landmarks
- Education

#### 2.2.2 Visual presentation

Data gathered will be presented to the user in a visually attractive user interface which will make the user experience enjoyable. Several modules will be used to construct a modern, stylish and useful interface:

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

- Minimal textual information (the user will not be burdened by volumes of text)
- Images, photos
- Maps (tagged with all the most important locations)
- UI gadgets (graphs, charts and more)

### 2.2.3 Preferences system

In order for the application to help the user decide where he would like to go to study abroad some information will need to be provided by the user. There will be several different criteria on which the user will be able to provide information which combined will allow the application to present a list of recommended locations to the user. Some of the criteria follow:

- Country / Language: One of the most important factors is the destination country and also the language spoken there. Some students choose countries where the language is completely unknown to them, and some prefer countries with familiar languages (perhaps the native language or some language learned during education).
- University: Another important factor is the university they plan to study abroad at. The ranking and reputation of the university is important and also the study programs they offer.
- City: The city of the university is very important. Is it a metropolis or a small town, is it seaside or continental, prices of common items, monthly expenditure, climate, safety, social life score. All of these are factors the user will be able to input in order to get a city to his preferences.
- Financing options: Possibility of scholarships, student loans, part time work.

### 2.2.4 Recommendation system

The application will gather information about the user during his time spent using the application in order to provide additional suggestions, hints and other types of feedback designed specifically for the user.

## 2.3 Project milestones

Project goals will be accomplished through several milestones. In general the milestones are:

- Project vision
- Project plan
- Requirements definition
- System design
- Alpha prototype
- Beta prototype
- Acceptance test
- Release candidate
- Final product

The alpha and beta prototype will contain additional application specific milestones such as: database definition and implementation, acquiring data sources, recommendation system development, user interface mockup, core keywords functionality, user reviews and comments, scientific functionalities, Google maps interaction and more.

## 2.4 Project deliverables

General project milestones will be followed by several project deliverables. The main part of the deliverables is the documentation that sets grounds for future development such as:

- Project plan document
- Requirements definition document
- Design description document

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

Besides documentation the milestones will be accompanied by presentations of work accomplished so far. There will be six presentations in total:

- Team and Project vision presentation
- Project plan presentation
- Requirements Definition and System Architecture presentation
- Alpha prototype presentation
- Beta prototype presentation
- Final project presentation

Finally, the last deliverable will be the final product delivered with installation setup and source code.

## 2.5 Project testing

Project testing will be done in cooperation with the project supervisor and will produce the acceptance test plan document. Acceptance tests will be performed for detection of possible defects and a test report will be published. Testing manager will be in charge of the test team and the testing campaign.

## 2.6 Product delivery

After project testing a final project report document will be published and all the other documents (such as various technical documents, user manuals, etc.) will be finalized. The packaged product (together with installation and source code) will be delivered to the project web page.

## 3. Organization

### 3.1 Project management

- Project home: Zagreb, Croatia (FER)
- Project supervisor: Ivana Bosnić (FER)
  
- Project leader: Branimir Lochert (FER)
- Team leader: Alessandro Sisto (PoliMi)

The project leader Branimir Lochert is also the team leader for the Croatia (FER) local team.

Project management is responsible for guiding the project towards its goals by distributing work to all team members and meeting project deadlines which includes accomplishing project milestones and producing the required deliverables on time. Team leaders are responsible for their local team while the project leader is responsible for the whole project and communication with the project supervisor.



StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

### 3.2 Project group

The project team is distributed with three team members from Milano, Italy attending PoliMi and three team members from Zagreb, Croatia attending FER. Explicit list of all project team members and their responsibilities follows:

Name	Initials	Responsibility (roles)
<b>Branimir Lochert</b>	BL	Project leader, Documentation manager, Risk manager, Developer (Server)
<i>Alessandro Sisto</i>	AS	Team leader, Backup manager, User interface manager, Developer (Client)
Katarina Sekula	KS	Requirements manager, Testing manager, Developer (Server)
Milan Čop	MČ	Lead Developer, Database manager
Daniele Rogora	DR	System architect, Virtual Machine manager, Developer (Client)
Javier Hualpa	JH	SVN Manager, Data sources manager, Developer (Server)

Some responsibilities and roles may change during the project.

### 3.3 Project responsibilities

#### 3.3.1 Project leader (manager)

The project leader (manager) responsibilities include planning, organizing, managing, leading and controlling the project and project team members for the purpose of achieving all of the project goals and objectives inside the preconceived deadlines.

#### 3.3.2 Team leader (manager)

The team leader (manager) responsibilities include providing guidance, instructions, directions and other forms of management to the assigned local team. The team leader reports the status and progress of his team to the project leader.

#### 3.3.3 Documentation manager

The documentation manager reviews and edits all written documentation before it is published to make sure it conforms to documentation policies specified for the project.

#### 3.3.4 Risk manager

The risk manager advises the team on any potential risks to the project. He/she identifies and assesses threats to the project and prepares contingency plans in case things go wrong.

#### 3.3.5 Backup manager

The backup manager makes sure all the production material is backed up on regular intervals. In case of any data loss he is responsible for the recovery process.

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

### 3.3.6 *User interface manager*

User interface manager has a primary role in user interface design and implementation. He/she also makes sure that the user interface is ergonomic and accessible to the user.

### 3.3.7 *Requirements manager*

Requirements manager is the person responsible for gathering requirements from communication with relevant stakeholders. Additionally he/she is responsible for the documentation, analysis, tracing and prioritizing gathered requirements.

### 3.3.8 *Lead developer*

Lead developer's responsibilities are to manage items and tasks his developing team is supposed to accomplish in the current iteration. He/she is to provide guidance and assistance in case a member of his team is stuck on a certain task. Additionally he is to make sure that his team is adhering to the software quality standards agreed by the project team.

### 3.3.9 *Testing manager*

Testing manager is responsible for leading the test team and effectively implementing a testing process. The testing manager must create a test plan and execute the test campaign described in the plan. Finally the testing manager is responsible for producing test reports and metrics and assuming the role of the quality assurance manager.

### 3.3.10 *Database manager*

Database manager is expected to determine the best possible method of organizing and storing data and then implementing that method. After implementation he/she is responsible for maintaining the database.

### 3.3.11 *System architect*

System architect's responsibilities are to design the architecture of the system on an abstract level by defining core features and elements which provide the framework for accomplishing project requirements.

### 3.3.12 *Virtual machine manager*

Virtual machine manager is in charge of setting up and maintaining a development environment for the project team. He/she controls which software is installed on the virtual machine.

### 3.3.13 *SVN manager*

The SVN manager is in control of the project SVN repository. His/hers responsibilities are to determine the project SVN policies and to enforce them in the repository.

### 3.3.14 *Data sources manager*

The data sources manager is responsible for collecting and organizing open sources of information found on the web which are required for the project goals. His/hers responsibilities also include studying ways to interface with the data sources.

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

### 3.4 Project tools

Project tools used:

- Management: MS Project
- Collaboration: Google groups, Planbox, SVN, Gliffy

### 3.5 Customer

Customers: Prof.dr.sc. Mario Žagar (FER), Prof. Raffaella Mirandola (PoliMi)

Students, postdocs and generally people who wish to study abroad are the external customers/users for the project. Surveys will be performed targeting this group of people to find out what are some of the important criteria students use to decide where they want to go to study abroad.

### 3.6 Supervisor

Project supervisor is Ivana Bosnić (FER).

### 3.7 Others

Marin Orlić (FER) – virtual machine coordinator.

All of the DSD course staff from all three of the countries: Croatia (FER), Italy (PoliMI) and Sweden (MDH).

## 4. Development process

### 4.1 Overview – Personal RUP

Strict deadlines for the first deliverables have been defined, as shown in Section 5. Thus the group doesn't have many options available for the development of the first phases of the project; indeed, to be able to present a Design Description Document in time it will be necessary to focus heavily on the whole system architecture and on the requirements prior to any other kind of activity.

This leads to a preliminary feasibility study followed by requirements specification and design of the system. The nature of our project, that is prone to risks and that tries to produce something innovative, on the other hand, makes it very difficult to really identify and analyze these requirements in the first phases. Thus in the implementation phase the group will follow an iterative approach.

The requirements and design documents will be followed as much as possible, but, should the requirements change, the group will be able to revise them, as deeply as needed.

The Rational Unified Process will be used as a reference for the development: small and short iterations will focus on relatively small system modules, as in the elaboration phase of the RUP. In this way risks related to the information sources will be addressed immediately, and also preliminary unit testing will take place during the implementation phase.

The picture represents the usual RUP development plan, showing also the effort of a particular kind needed in each of the phases. It is worth noting that analysis and design has a greater importance in our project than in the usual RUP because of the deadlines for the related documents that we must respect.

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

## Detailed process

### 4.1.1 Inception phase

All team members build a shared project vision, talking with the customer and adding new ideas, exploring the web sources and interviewing possible stakeholders. The artifact produced is a set of documents built with the whole team cooperation.

Specifically, these documents are the Requirements Definition document, describing the requirements for the project and possible use cases, and the Design Description document, that contains a high level description of the system, its components and their interactions as well as the technologies adopted for the purpose, in terms of frameworks and programming languages, COTS, hardware interfaces and so on.

### 4.1.2 Elaboration phase

In this phase most of the requirements instability will be targeted. It will be the core of the implementation of the project's critical modules. Different local teams will work concurrently and iteratively on different system modules. The division will be on server and client side modules, that means on the background service and the user interface.

Each iteration will have a fixed time length and will result in a working and tested piece of product, that may be either a new module or an enhanced version of an existing one, ready for evaluation by the team members. Also specific documentation should be generated during each iteration related to the work done within it.

The phase will have as a final artifact an alpha version of the project. This means that all the requirements and features will have been settled as well as all critical system modules.

### 4.1.3 Construction phase

The main tasks of this phase are integration and testing. No new sources should be added during this phase. Again, an iterative approach is taken in this phase; each short and time fixed iteration produces a working system snapshot. Final artifacts should be a working beta of the product and an acceptance test plan, defined in collaboration with the customer.

### 4.1.4 Transition phase

Final testing, polishing and acceptance are the goals of this phase. The artifacts produced are a deployable package containing the whole system software, a package containing its source code, the user documentation and a final project report.

## 4.2 Roles

Reference: <http://www.ibm.com/developerworks/rational/library/apr05/crain/>

An iterative process like RUP defines two kinds of roles: breadth and depth roles. The former kind is such that must focus on the overall process within each iteration; it has a wider perspective, and drives and adjusts the overall solution. The latter instead must focus more deeply on details of a smaller piece of the project within each iteration.

In our project team that is composed of six members, we don't plan to embrace this separation between breadth and depth roles, but each team member is required to always have an updated view of the overall project, apart from focusing on his specific task.

Indeed every role detailed in section 3.3 is a breadth role, since it is project-wide, i.e. it has responsibility for the whole project. Also each team member will also have the depth role of developer.

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

### 4.3 Progress tracking

Project progress reports are written weekly by each team member.

### 4.4 Quality assurance

Apart from periodic testing during each iteration of the elaboration and construction phases, a constant customer involvement is prospected to keep the project always aligned with the customer desires.

The quality assurance manager will constantly monitor the project and check that it meets the quality standards defined by the group and the customer.

## 5. Deliverables

To	Output	Planned week	Promised week	Late +/-	Delivered week	Rem
Customers/Project supervisor	Project plan document	43	44	1	44	
Customers/Project supervisor	Requirements definition document	44	44	0	44	
Team Members (internal)	SVN Policy Document	44	44	0	44	
Customers/Project supervisor	Design description document	45	45			
Team Members (internal)	Coding Conventions Document	45	45			
Team Members (internal)	Domain model class diagram (Alpha Milestone)	45	45			
Team Members (internal)	Domain model c# classes – source code (Alpha Milestone)	45	45			
Team Members (internal)	Data sources DS1, DS2 report	45	45			001
Team Members (internal)	C# classes for parsing /scraping data source DS1 - source code (Alpha Milestone)	46	46			
Team Members (internal)	C# classes for interfacing with data source DS2 - source code (Alpha Milestone)	46	46			
Team Members (internal)	C# classes for dynamic loading framework (Alpha Milestone)	46	46			
Team Members (internal)	C# classes for server interface to client (Alpha Milestone)	47	47			
Team Members (internal)	JavaScript code for client interface to server (Alpha Milestone)	47	47			
Team Members (internal)	Complete GUI modules source code (Alpha Milestone)	48	48			
Team Members (internal)	Complete backend modules source code (Alpha Milestone)	48	48			

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

Customers/Project supervisor	Alpha prototype (Browsing universities by location) – Complete source code	48	48			
Team Members (internal)	Database ER diagram (Beta Milestone)	48	48			
Team Members (internal)	Database creation SQL script (Beta Milestone)	48	48			
Team Members (internal)	Database file (Beta Milestone)	48	48			
Team Members (internal)	C# classes for a data access layer (Beta Milestone)	48	48			
Team Members (internal)	Log In / Registration C# classes source code (Beta Milestone)	48	48			
Team Members (internal)	Log In / Registration GUI source code (Beta Milestone)	48	48			
Team Members (internal)	Data sources DS3, DS4 report	48	48			002
Team Members (internal)	C# classes for parsing /scraping data source DS3 - source code (Beta Milestone)	49	49			
Team Members (internal)	C# classes for interfacing with data source DS4 - source code (Beta Milestone)	49	49			
Team Members (internal)	Recommendation system C# classes (Beta Milestone)	50	50			
Team Members (internal)	Recommendation system GUI source code (Beta Milestone)	50	50			
Team Members (internal)	Google maps API JavaScript source code (Beta Milestone)	50	50			
Team Members (internal)	Postdocs recommendation system C# classes (Beta Milestone)	50	50			
Team Members (internal)	Postdocs recommendation system GUI source code (Beta Milestone)	50	50			
Team Members (internal)	Complete GUI modules source code (Beta Milestone)	51	51			
Team Members (internal)	Complete backend modules source code (Beta Milestone)	51	51			
Customers/Project supervisor	Beta prototype (Recommendation system) – Complete source code	51	51			
Team Members (internal)	Commenting system C# classes (Final Milestone)	52	52			
Team Members (internal)	Commenting system GUI source code (Final Milestone)	52	52			
Team Members (internal)	User profile system C# classes (Final Milestone)	52	52			
Team Members (internal)	User profile system GUI source code (Final Milestone)	52	52			
Team Members (internal)	Additional data sources report	52	52			

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

Customers/Venture capitalist	Acceptance test Plan	52	52			
Team Members (internal)	Advanced recommendation system C# classes (Final Milestone)	01	01			
Team Members (internal)	Advanced recommendation system GUI source code (Final Milestone)	01	01			
Team Members (internal)	Complete GUI modules source code (Final Milestone)	02	02			
Team Members (internal)	Complete backend modules source code (Final Milestone)	02	02			
Customers/Venture capitalist	Test report	03	03			
Customers/Venture capitalist	Revised version of all Documents	03	03			
Customers/Venture capitalist	Final product (with Social System) – Complete source code	04	04			

### 5.1.1 Remarks

Remark Id	Description
001	DS1 – Data Source 1 : <a href="http://www.4icu.org/">http://www.4icu.org/</a> , DS2 – Data Source 2 : <a href="http://en.wikipedia.org/">http://en.wikipedia.org/</a>
002	DS3 – Data Source 3 : <a href="http://www.expatistan.com/">http://www.expatistan.com/</a> , DS4 – Data Source 4: <a href="http://www.mendeley.com/">http://www.mendeley.com/</a>

## 6. Inputs

From	Required item	Planned week	Promised week	Late +/-	Delivered week	Rem
Customers/Project supervisor	Project Vision feedback	43	43	0	43	
Customers/Project supervisor	Project Plan feedback	44	44	0	44	
Customers/Project supervisor	Requirements and System Design feedback	45	45			
Customers/Project supervisor	Alpha Prototype feedback	48	48			
Customers/Project supervisor	Beta Prototype feedback	51	51			

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

External testers	Usability Reports	01	01			
External testers	Testing Reports	02	02			

## 7. Project risks

Dimension	Possibility	Risk	Preventive action
Requirements	High	Requirements inflation	Constantly involve the Project supervisor. Determine a list of frozen requirements that will be delivered and then evaluate whether the schedule allows new ones.
Planning & Control	High	Unrealistic schedules	Estimate tasks carefully. Meet internal deadlines established by the Project leader. Let the team know if you are unable to complete a task so it can be rescheduled or distributed.
Team	Medium	Lack of technical skills	Identify among the team members the one that is most proficient with the skill as the mentor to the team. If no team member is proficient enough with the required skill then one team member should be responsible for developing the skill and then provide fast track mechanisms/materials for the rest of the team to acquire it.
Planning & Control	Medium	Poor productivity	Have short iterations, this way work is time boxed into a manageable iteration(5 days, 1 week, no more than that) and there is always a sense of urgency.
Technology	Medium	SVN issues	Adhere to the SVN policy when working with documents or code. The SVN manager should perform the appropriate backups.
Planning & Control	Medium	Ineffective communications	Schedule formal meetings at least once a week. Express any concern in the Google group set for it. Contact any team member via email or Skype.
Technology	Low	Incorrect technology choice	Have a plan B in case the chosen technology is not suitable for the project.
Team	Low	Personnel shortfall	Assign workload based on preferences/skills. Co-locate related workload. In case any team member is unavailable, distribute the workload among other team members.



StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

## 8. Communication

Communication in the team is divided in two parts:

- The local communication
- Communication of the whole team.

All communication will be in English, except for communication essential for the local team. In this case, communication can be in Croatian / Italian.

### 8.1 Synchronous communication

- Local meetings  
Meeting if needed - Communication between local team members who discuss about problems or meet with the supervisor.
- Skype  
Weekly meeting - The whole team meets Thursdays at 18:00.  
Meeting if needed - Communication between team members who work together on a task.

### 8.2 Asynchronous communication

- Email  
For a discussion between one or more members of the team
- Google group  
For discussion and information sharing between team members or communication between team members and supervisor
- Planbox  
Allows a member of one team to collaborate in real-time with entire team, whether they are distributed or collocated.

## 9. Configuration management

- SVN (Subversion) will be the repository for code and project documentation.
- The SVN policy document provides detailed information about coordinated usage and preventive actions to take.
- Google Docs and Google Groups will be the tools used for drafting the documents when they are initially created.
- The SVN manager and the Documentation manager will handle any corrective action regarding merging or lack of content in code and documents respectively.

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

## 10. Project plan

### 10.1 Time schedule

Id	Milestone Description	Responsible Dept./Initials	Finished week			Metr.	Rem.
			Plan	Forecast Week +/-	Actual		
/	---Inception phase---						
M001	Project vision	All	42	42	0	42	0
/	---Elaboration phase---						
M002	Project plan	BL	43	43	0	44	-1
M003	Requirements Engineering	KS	44	45	1		
M004	System Design	DR	45	45	0		
/	---Construction phase---						
/	---Alpha prototype---						
M005	Application domain model	BL, MČ, KS, JH	45	45	0		001
M006	Interfacing/parsing data sources #1	BL, MČ, KS, JH	46	46	0		002
M007	Dynamic loading framework	BL, MČ, KS, JH	46	46	0		003
M008	Server – client interface	All	47	47	0		004
M009	GUI Modules/Mockups #1	AS, DR	48	48	0		005
/	---Beta prototype---						
M010	Database definition	BL, MČ, KS, JH	48	48	0		006
M011	Data Access Layer	BL, MČ, KS, JH	48	48	0		007
M012	Log in/Registration system	All	48	48	0		008
M013	Interfacing/parsing data sources #2	BL, MČ, KS, JH	49	49	0		009
M014	Recommendation system	All	50	50	0		010
M015	Interacting with Google maps	AS, DR	50	50	0		011
M016	Postdocs recommendations	All	50	50	0		012
M017	GUI Modules/Mockups #2	AS, DR	51	51	0		013
	---Transition phase---						
M018	Commenting system	All	52	52	0		014
M019	User profile system	All	52	52	0		015
M020	Interfacing/parsing data sources #3	BL, MČ, KS, JH	52	52	0		016
M021	Advanced recommendation system	All	01	01	0		017
M022	GUI Modules/Mockups #3	AS, DR	02	02	0		018
M023	Documentation	BL	03	03	0		
M024	Testing	All	03	03	0		
M025	Final delivery	BL	04	04	0		

#### 10.1.1 Remarks

Remark Id	Description
001	Developing C# domain model classes to abstract data gathered from data sources (University, City, etc.)
002	Parsing/scraping data source DS1; Interfacing with data source DS2 API
003	Developing a set of C# classes for loading data only on demand
004	Developing a set of C# classes and JavaScript functions to interface between the client and the server
005	GUI modules for browsing the universities by location, for displaying university and location information. Also a basic page layout and style

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

006	Database model definition and implementation
007	Developing a set of C# classes to interface with the database
008	GUI and server modules for enabling user registration and login
009	Parsing/scraping data source DS3; Interfacing with data source DS4 API
010	GUI and server modules for a recommendation system
011	JavaScript functions for interacting with Google maps API
012	GUI and server modules for a recommendation system postdocs module
013	GUI modules for preferences questionnaire, registration forms, login dialogs, Google maps holder and controls
014	GUI and server modules for a commenting system to comment universities and locations
015	GUI and server modules for a user profile system
016	Interfacing/parsing any additional data sources needed
017	Improving the recommendation system, editing recommended universities list, editing preferences
018	GUI modules for user profile page, commenting sections, recommended for you section

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

## 10.2 Activity plan

	w40	w41	w42	w43	w44	w45	w46	w47	w48	w49	w50	w51	w52	w01	w02	w03	w04
Project vision			Blue														
Project plan			Red	Red	Red	Red											
Requirements Engineering			Red	Red	Red												
System Design				Red	Red												
Application domain model					Light Green												
Interfacing/parsing data sources #1					Light Green	Green											
Dynamic loading framework						Green											
Server – client interface						Green	Green										
GUI Modules/Mockups #1							Green	Green									
Database definition								Light Purple									
Data Access Layer								Light Purple									
Log in/Registration system								Light Purple									
Interfacing/parsing data sources #2								Light Purple	Purple								
Recommendation system									Purple	Purple							
Interacting with Google maps									Purple	Purple							
Postdocs recommendations										Purple							
GUI Modules/Mockups #2										Purple	Purple						
Commenting system											Light Blue	Blue					
User profile system											Light Blue	Blue					
Interfacing/parsing data sources #3												Blue	Blue				
Advanced recommendation system													Blue	Blue			
GUI Modules/Mockups #3														Blue	Blue		
Documentation			Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey
Testing			Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey	Grey
Final delivery																	Orange
Iterations:				I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12		
Prototypes:							Alpha	Alpha	Alpha	Beta	Beta	Beta					
Phases:	Inception	Inception	Inception	Elaboration	Elaboration	Elaboration	Construction	Construction	Construction	Construction	Construction	Transition	Transition	Transition	Transition	Final	Final

StudyAbroad	Version: 1.00
Project Plan	Date: 2012-11-02

ID	Predecessor	Activity	Days	Mdays	Rem.
/		---Inception phase---			001, 002
M001		Forming project vision	6	36	
/		---Elaboration phase---			
M002	M001	Project planning	5	30	
M003	M001	Requirements Engineering	5	30	
M004	M003	System Design	6	36	
/		---Construction phase---			
/		---Alpha prototype---			
M005	M004	Developing application domain model	3	12	
M006		Interfacing/parsing data sources #1	4	16	
M007	M005	Developing dynamic loading framework	4	16	
M008	M004	Server – client interfacing	4	16	
M009		GUI Modules/Mockups production #1	8	16	
/		---Beta prototype---			
M010	M005	Database implementation	1	4	
M011	M010	Developing a data access layer	2	8	
M012	M011	Developing a log in/registration system	2	12	
M013		Interfacing/parsing data sources #2	6	12	
M014	M005, M006, M007, M012, M013	Developing a recommendation system	12	24	
M015		Interacting with Google maps	6	12	
M016	M014	Developing postdocs recommendations system	6	12	
M017		GUI Modules/Mockups #2 production	10	20	
/		---Transition phase---			
M018		Developing a commenting system	2	12	
M019	M012	Developing a user profile system	2	12	
M020		Interfacing/parsing data sources #3	2	8	
M021	M014	Developing an advanced recommendation system	7	28	
M022		GUI Modules/Mockups #3 production	7	14	

Total planned calendar days for the project to be finished and total planned man-days required to finish the project:

Planned effort (days)	Planned effort (man-days)
91	386

### 10.2.1 Remarks

Remark Id	Description
001	4-hours working days supposed
002	Time spent on documentation/testing of each item is included