

Project Portfolio for Upwind::Spring 2012

Anu Pramila, 1685372, apra@ee.oulu.fi
Andrei Vainik, 1961863, antti.vainik@gmail.com
Juha-Matti Hurnasti, 1833836, hurnasti@paju.oulu.fi
Tomi Sarni, 1542473, tomi.sarni@gmail.com

May 22, 2012

Abstract

UpWind project is a mainly student run software project which started in 2006. Each year two project teams are formed to tackle the backlog items set by UpWind project leaders. This document summarizes achievements and experiences of the Spring 2012 project team. The project teams are formed mainly from students taking part the Project 2 -course.

The UpWind project is about building a sailboat navigation software using C++ language in Qt development environment. Currently the UpWind architecture is going through a complete overhaul as the new plugin based architecture was introduced in 2010. The aim for Spring 2012 project team was to import and integrate missing route calculation parts from the old system to the new one. Although the team was not able to fully achieve this task, much needed documentation for the future project teams was done.

Table of Contents

	page
<u>Abstract</u>	
<u>1 Project Results</u>	4
<u>2 Project Planning</u>	5
<u>3 Project Work and Management</u>	6
<u>4 Steering Group Meetings</u>	8
<u>5 Appendices</u>	9

1 Project Results

UpWind is an Open Source Software project initiated and coordinated by the UpWind team at Department of Information Processing Science at University of Oulu. The aim of the project is to build an advanced navigation software for sailboats. Since 2006 several project teams have contributed to the project improving and adding features to the software. However, as the code has become difficult to maintain and understand, a new plugin based architecture was introduced. Spring 2010 project team designed and reimplemented many of the Upwind navigator software according the new plugin architecture. Fall 2011 team ported most of the remaining software components to the new architecture.

The plan of the Spring 2012 project team was to port the remaining Math-plugin to the architecture. This plugin includes functionalities of automatic long term and short term route planning. The old architecture contained these functions and so according to the plan the work was assumed to be mostly porting of the old code to the new architecture style.

However, after the first three sprints, a choice was made to focus solely on long term route planning and use the remainder of the time to make a documentation of the project for the future teams. The documentation now includes guides on how to compile the old and new codes, how to use version control system, where to download required software and what difficulties our team faced and how they were overcome. There are also some improvement suggestion included in the document.

In the end of the project, the long term route planning was completed and documentation readied for the future teams so that they could work more efficiently from the beginning. There have been made also some optimizations to the code created by the previous project groups. The aim of the optimizations have been to enable the program itself to launch faster and to make it more efficient.

2 Project Planning

In the beginning of the project a project plan was constructed. The project was designed to have four sprints, each sprint contributing towards the final product. The aim of the project was quite clear from the beginning, but it was difficult to estimate the required workload without being familiar with the already existing code. As the work progressed, we were able to see how much we had completed and in the end how much work we had left. The planning and controlling of our efforts was done between sprint with scrum meetings. The notes of these meetings have been included in the end of this document.

Due to the nature of the problem, planning for the project was kept at minimum. First, in order to plan effectively, we needed to know more about the problem at hand. Therefore the first sprint was dedicated in familiarizing ourselves with the old and new codes. During this process we found that many features we thought were implemented in the new code, were in fact not. We also found that everything had not been implemented perfectly according to the plugin architecture.

After we had familiarized ourselves with the work of previous project teams, we selected few goals for our work, that is, short and long term route planning. We knew that in the old code the responsibility of calculating the route was dedicated to multiple classes and shared among many files. Therefore we understood that the work was not straightforward porting of old code to the new and would require lot of time.

Beginning the work and writing the first lines of code was difficult but by the end of the second sprint we had written code towards the long term route planning and the goals seemed achievable. However, in the end of the third sprint we struggled with a coding issue and were forced to reevaluate the goals for the last sprint.

During the the whole project one single obstacle or problem have been faced several times by each member of the project group. This was that the former groups working in this project have not made the documenting work properly, or in some cases, have not done it at all. This has caused a great deal of frustration to the group and probably have also lowered the overall moral of the group and interest towards the project itself. To prevent this happening to the next group starting with UpWind project in next fall, we decided to write a project manual which would consist of tips, hints and instructions to the next group to make their project launch easier than it was for us.

3 Project Work and Management

In the project plan, the project was divided into four sprints. The aim of the first and the shortest sprint was to make the project plan and get familiar with the working environment. This goal was reached well.

The aim of the next sprint was to get familiar with the code and locate and identify pieces of code that are necessary for long term route planning. In addition it was planned that some coding is done in the end of the sprint. This goal was achieved as well.

The third sprint requirement was to code and test long term route planning, and in the end, complete it and prepare for the short term route planning. However, although the code was successfully ported to the new architecture, there remained some problems that needed to be solved before moving to the next task. It was agreed in the Steering Group Meeting that the goal of the final sprint would need revision. It was noted that understanding the code is a long process and could be improved with proper documentation.

Therefore, the goal of the fourth sprint was updated and a choice was made to focus solely on long term route planning and use the remainder of the time to make a comprehensive documentation of the project.

None of the problems faced during the project were overwhelming or halting. Some of the features were not implemented as assumed and there were some design issues that required time to solve. However, the fact that there was no documentation neither of the old code nor the new code slowed the implementation process considerably.

Because there were no documentation, there was hardly any information about the formats of data received from the database, nor in which format the data should be processed. Everything had to be examined and understood through reverse engineering.

In order to easily share information and results of the aforementioned reverse engineering, the project team worked at the same premises in the classroom FY1052 at the university of Oulu. In addition, Google Docs was taken advantage of and all the documents needed and created were stored and modified in there. Redmine system was used in order to store Scrum meetings and Steering Group minutes as well as keeping record of Git version control system activities.

Even in the beginning of the project, some time was allocated getting familiar with the system and planning the project. It was well known that the most limiting aspect of the project was time. Each of the four project team members had 300 hours to use for the project and the project should be planned accordingly. Therefore, only limited number of goals were chosen for the project and enough time was reserved for solving problems should any occur during the project.

Because all the team members worked at the same location, there was no need for arranged and timed meetings and therefore the scrum meetings were held when all the team members were present at the same time and some update of the state of the project seemed useful. For the same reason, no project manager nor Scrum master was chosen as such but the responsibilities circulated among the project members according to the time and issue at hand. This circumstance had an effect that each of the project members were reasonably familiar with any aspect of the code. Although some of the project members were more

like experts of some parts of the code, others had enough knowledge to understand how changes in expert's code might affect theirs.

All the problems in the code were discussed as soon as possible when they appeared and solved through discussions on the classroom or by e-mail and chat. The most notable problems included understanding of the code and some bugs in the code. It was feared in the beginning of the project that sickness or other personal matters of any of the team members would slow down or even increase the workload of other members. Fortunately, we were able to divide tasks on such a manner that this risk did not finalize.

Each of the team members knew the deadline of the next Steering Group meeting and requirements for the meeting were discussed always on the previous week before the meeting. A presentation was created for the meeting and invitations sent through project mailing list. The mailing list was also used if there were any issues or problems during a sprint that either occurred when all the project members were not available in the classroom or the answer required an opinion of TOL representative Samuli Saukkonen or Project coordinator Víctor Arroyo.

Final report of the project is distributed along this document as an appendix.

4 Steering Group Meetings

Date and time for the next Steering Group Meeting was often selected in the end of previous meeting. Sometimes the date was discussed later by e-mail but after the date had been decided and agreed by all the meeting participants the official invitation was sent by e-mail two days before the meeting. This was deemed more than sufficient by the TOL representative and Chairman Samuli Saukkonen. The invitation included possible documentations for reviewing before meeting.

Steering Group meeting minutes have been included as an appendix in the end of this document. The meetings were held in FY1052 at the university of Oulu. The meetings did not follow a strict procedure although the meeting invitation contained a schedule which was held as a guideline.

After each of the participants had arrived, the meeting was started. In the beginning, the names of each one present were documented. Then the project team presented the results of the latest sprint with a slide show, including hours spent for the project. The status of the project were then discussed along with proposals for the next sprint. Finally the decisions were made either to continue as presented in the project plan or to make changes to the plan. The date for the next meeting was tentatively agreed in the end of the meeting, but sometimes it was changed later by e-mail.

5 Appendices

Appendix 1: Steering group meeting 1 : 16.2.2012 - Meeting minutes

There will be two days time to make comments this document.
Meeting started 16.55

Participants:

Samuli Saukkonen (Chairman)
Tomi Sarni (Secretary)
Anu Pramila
Juha-Matti Hurnasti
Andrei Vainik

Issue 1 - Opening the meeting

Meeting has started without Victor Arroyo.

Issue 2 - Project Review

2.1 Status of the project.

The development team feels that they are stuck with the project at the moment. There are problems understanding the architecture.
It appears that there are parts of the code that are unfinished, which prevent/slow the work of the team concerning the given task.
Has the team got all the documents available, especially regarding the architecture?

So far the team has spent total of 220 hours.

Tomi 60 hours
Anu 60 hours
Andrei 60 hours
Juha-Matti 40 hours.

The morale is bit low with the team due to the overwhelming obstacles the team seem/feel to be facing.
Also Juha-Matti and Tomi still lack the keys to the work premises.

2.2 Review and discussion of project plan.

Samuli said that the project plan is good. Only alteration to be made will be the time when the material needs to be submitted.
Which was mentioned to be two weeks, instead it will be two days. Anu will make the change to the project plan.

Samuli said that pre-study report is good and there is no need for alteration.

2.3 Next phase of the project.

Andrei feels that the team will need more outside help with understanding the architecture, understanding some of the concepts.

Samuli promised to handle a date or time for this "Workshop".

2.4 Decision making

- Project plan and pre-study report accepted
- arrange a workshop with the team and outside help (Victor)
- get all the documents for the team
- keys for Juha-Matti and Tomi as soon as possible

3 Other issues

None.

4 Next Meeting

Time: 14.3.2012 16:30

Place: FY1052

5 Ending the meeting

Meeting ended at 17.30

Appendix 2: Steering group meeting 2 : 15.3.2012 - Meeting minutes

There will be two days time to make comments this document.

Meeting started 16.30

Participants:

Samuli Saukkonen (Chairman)
Victor Arroyo (Project Manager)
Tomi Sarni (Secretary)
Anu Pramila
Juha-Matti Hurnasti
Andrei Vainik
Aruna Choragudi

Issue 1 - Opening the meeting

The meetings has been opened.

Issue 2 - Project Review

2.1 Status of the project.

The team presented the current state of the project with slides and demo of the system.

- Long from route planning imported from old code
- Route draw renderer started

So far the team has spent total of 500 hours.

Tomi 160 hours
Anu 120 hours
Andrei 130 hours
Juha-Matti 88 hours.

2.2 Review and discussion of project plan.

All feel the sprint plans are quite good and no need to deviate from them.

2.3 Next phase of the project.

Start sprint 3.

2.4 Decision making

Decision were made to add more emphasis on documentation in the last phase of the project. (Sprint 4).

3 Other issues

No other issues.

4 Next Meeting

Time: 3.4.2012 16:30
Place: FY1052

5 Ending the meeting

Aruna presented her Master's Thesis considering the UpWind Project.

Meeting ended at 17.21

Appendix 3: Steering group meeting 3 : 3.4.2012 - Meeting minutes

There will be two days time to make comments this document.

Meeting started 16.30

Participants:

Samuli Saukkonen (Chairman)

Tomi Sarni (Secretary)

Anu Pramila

Juha-Matti Hurnasti

Andrei Vainik

Aruna Choragudi

Issue 1 - Opening the meeting

Meeting has started without Victor Arroyo.

Issue 2 - Project Review

2.1 Status of the project.

There has been considerable problems with long term route planning, especially with coordinate transformations and debugging routes.

The long term route planning now can plot a course on shortest possible. no weather added. some values missing from DB.

Tomi 220 hours

Anu 160 hours

Andrei 150 hours

Juha-Matti 128 hours.

2.2 Review and discussion of project plan.

Team and Samuli agreed that the remaining focus would be on long term route planning and producing some learning documents for next group.

2.3 Next phase of the project.

To finalize long term route planning.

2.4 Decision making

3 Other issues

Samuli asked about the project portfolio and the team set that it would be done for last SGM meeting. Aruna presented her work regarding the changing wind conditions simulation.

4 Next Meeting

Time: 22.5.2012 16:30

Place: FY1052

5 Ending the meeting

Meeting ended at 17.15

Appendix 4: Steering group meeting 4 : 22.5.2012 - Meeting minutes

There will be two days time to make comments this document.

Meeting started 16.30

Participants:

Samuli Saukkonen (Chairman)
Victor Arroyo (Project Manager)
Tomi Sarni (Secretary)
Anu Pramila
Juha-Matti Hurnasti
Andrei Vainik

Issue 1 - Opening the meeting

Issue 2 - Project Review

Jussi Hurmasti presents the slides and the status of the project.

2.1 Status of the project.

Currently the long term route planning is working. A lot of optimization have been done to the chart controls and route plotting. Chart graphics have been improved. Also the system is connected to a simulator and data is being showed in the system. The support documentation has also been done for the future project groups. New Git repository have been created where all the latest source code has been put for next team

Short term planning was not completed as agreed in SGM3.

2.2 Review and discussion of project plan.

Samuli Saukkonen felt that the project was a success and expected progress was made.

3 Other issues

Samuli wanted some feedback from the team. It will be added in final report.

4 Ending the project

Hours have been met and project is now concluded

5 Ending the meeting

Meeting ended at 17:07

Appendix 5: An example of steering group meeting invitation

UpWind	Meeting invitation				
Prepared Anu Pramila	File Asiakirja			Version 1.0	
Approved	Created TOL	Last saved 30.3.2012	Printed		

Project steering group meeting invitation

Time::	3.4.2012 at 16:30
Place:	University of Oulu FY1052
Delivery:	The project team, Victor Arroyo, Samuli Saukkonen

Agenda:

1. Opening the meeting
 - meeting participants
 - quorum status
 - selecting the secretary for the meeting
2. Approving the agenda
3. Project review:
 - The status of the project
 - Review and discussion of the status of the project according the Project plan
 - Next phase of the project
4. Decision-making (summary of decisions and actions points decided in this meeting)
5. Other issues
6. The next meeting
7. Ending of the meeting

Project team

Appendix 6:Daily Scrum Meetings

6.2.2012

What have you done since last meeting?

- Antti (Scrum master). Explored QT Plugin mechanism.
- Jussi. Compiled old codebase.
- Tomi. Try to get Plugins to load in new code version.
- Anu. Compiled old codebase. Made a first Steering group meeting invitation.

Did you have any impediments?

- Antti. Not really.
- Jussi. Path problem on Windows machine.
- Tomi. Path problem on Windows machine.
- Anu. No.

What are planning to do until the next meeting?

- Antti. Planning to get RouteManager working.
- Jussi. Explore Plugins mechanism.
- Tomi. Try to get Plugins to load in new code version. Prepare prestudy report.
- Anu. Explore old codebase and find documentation for new codebase.

7.2.2012

What have you done since last meeting?

- Antti (Scrum master). Familiarize himself with new architecture.
- Jussi. Familiarize himself with old class diagram / structure.
- Tomi. Try to get Plugins to load in new code version. Constructed a TestPlugin / familiarized himself about plugin creation.
- Anu. Explore old codebase and find documentation for new codebase.

Did you have any impediments?

- Antti. Yes. The UpWind server was down. This prevented him from running the application.
- Jussi. Yes. The UpWind server was down. This prevented him from running the application.
- Tomi. Yes. The UpWind server was down. This prevented him from running the application.
- Anu. Yes. The UpWind server was down. This prevented her from running the application..

What are planning to do until the next meeting?

- Antti. RouteManager
- Jussi. Continue exploring the Class diagram. Trying to reverse-engineer the old codebase.
- Tomi. Install Linux and the project on new upwind machine. Start working on math plugin.
- Anu. Start

8.2.2012

What have you done since last meeting?

- Tomi. Did not install linux as we managed to sort out the windows build problem. Installed QT and project on new machine. Started work on empty plugin which will be filled later by math.
- Antti. Started importing code from old Boat class to new version.
- Anu. Started writing the mandatory pre-study report
- Jussi. Installing the new machine. Making the class-diagram of the old codebase.

Did you have any impediments?

- Tomi. The new machines were password locked, managed to get password later.
- Antti. Unclearity about architecture. How are we supposed to implement communication between Plugins
- Anu. The subject is a bit unclear. Not much information found yet.
- Jussi. The correct version of Qt Creator was difficult to find.

What are planning to do until the next meeting?

- Tomi. Plan to finish the empty plugin and make it appear in the UI when core is run.
- Antti. Unclearity about architecture. How are we supposed to implement communication between Plugins
- Anu. Write introduction and some text for the pre-study report.
- Jussi. To get everything ready again.

13.2.2012

What have you done since last meeting?

- Antti. Trying to compile PostGRESql Plugin..
- Jussi. Set up work environment, browse the code, trying to understand architecture.
- Anu. Doing the pre-study report, trying to find related articles. Getting the code to compile.
- Tomi. Trying to understand architecture and set up work environment.

Did you have any impediments?

- Antti. Not enough understanding of the overlaying architecture.
- Jussi. Completely lost with the code. Not enough understanding of the architecture.
- Anu. Difficult to find relative information and articles.
- Tomi. Problems with windows environment.

What are planning to do until the next meeting?

- Antti. Draw a boat image on map.
- Jussi. Try to learn more about architecture.
- Anu. Continue work on pre-study report.
- Tomi. Decide what user stories to tackle.

16.2.2012

What have you done since last meeting?

- Jussi. Inspecting the code.
- Anu. Finished pre-study report and project plan, also sent those.
- Tomi. Continued work on importing long term route planning parts, specially trying to fetch necessary tables.

Did you have any impediments?

- Jussi. Everything sucks. Motivation low.
- Anu. couldn't connect to backlog, git not working. Code debugger problems.
- Tomi. Could not commit code, Git said "no permissions".

What are planning to do until the next meeting?

- Jussi. Trying to "break" the code.
- Anu. Trying to help Antti with coding. "pair programming".
- Tomi. Somehow get the necessary data for the route class. fake, direct DB call or through db plugin.

17.2.2012

What have you done since last meeting?

- Jussi. Been inspecting the code furiously. Took part in steering group meeting.
- Anu. Been inspecting the code.
- Tomi. Have tried to find how the current code fetches information from database and where it stores them. Also trying to get the plugin to load properly.

Did you have any impediments?

- Jussi. Still lacking a clear goal; what to do?
- Anu. Nothing special.

-Tomi. For some reason the chart is not loading in the current version (no errors). Git not working.

What are planning to do until the next meeting?

-Jussi. Trying to make one line of code that works and is useful.

-Anu. Continue inspecting the code.

-Tomi. Will start implementing some crude database fetches for bridges, transmission lines and navigation lines. Trying to get a pointer running through UpwindScene plugin part of the code.

23.2.2012

What have you done since last meeting?

-Jussi. Nothing.

-Anu. Familiarized myself with the code and starting to get a general idea what's happening there. Figuring out Git problems.

-Antti. Fixed Git user rights. Made an example graphics project, now understand how drawing is done in QT.

-Tomi. Trying to get necessary info from database to route.class.

Did you have any impediments?

-Jussi. No.

-Anu. Git has proven to be a bit difficult to use.

-Antti. not really.

-Tomi. Was away due flu for few days.

What are planning to do until the next meeting?

-Jussi. To find out what to do. Find own task.

-Anu. Start work on route widget.

-Antti. Start/continue work on boat widget.

-Tomi. Continue work on getting data from database for long term route planning.

5.3.2012

What have you done since last meeting?

-Antti. Tried to render a boat.

-Tomi. Finished importing and creating long term route planning.

-Jussi. Debugging long term route planning.

-Anu. Worked on RouteWidge.cpp and Inspecting renderer/drawing of objects and routes.

Did you have any impediments?

- Antti. Nothing special. Was away for few days due to flu.
- Jussi. Was a way for a week due to flu.
- Tomi. Due to lack of documentation, difficulties in understanding what sort of values and what meanings do some parameters and methods have in the program. Makes it hard to debug.
- Anu. THE GIT!

What are planning to do until the next meeting?

- Antti. Work on boat coordinate mapping.
- Tomi. Debug and test long term route planning.
- Jussi. Try to draw a test route on top of chart view.
- Anu. Try to get a working version of RouteWidget.cpp

6.3.2012

What have you done since last meeting?

- Antti. Compiled NMEA simulator and tried to connect it to uwcore.
- Tomi. Removed google based simulator and start work on importing the short term route planning parts.
- Jussi. Work on testing the long term route planning.
- Anu. Managed to get demo route drawn in screen with Route Widget.

Did you have any impediments?

- Antti. No.
- Tomi. No
- Anu. No.
- Jussi. No.

What are planning to do until the next meeting?

- Antti. Continue connecting NMEA simulator.
- Tomi. Continue work on short term route planning. Work on short term demo.
- Anu. Going to try to get NMEA simulator to work in mac.
- Jussi. Work on demo to get a route drawn over the chart.

13.3.2012

What have you done since last meeting?

- Antti. Tested NMEA simulator and UWCore on Linux/Windows platform.
- Tomi. Started work on short term route planning and some work on long term route planning integration.
- Anu. Managed to compile NMEA simulator but it did not work on mac. Merged

Tomi's code and planned how to connect route planning to UI.

Did you have any impediments?

-Antti. No.

-Tomi. Unable to use git from home. Not a serious problem.

-Anu. Was away few days due to a flu.

What are planning to do until the next meeting?

-Antti. Create a demo for steering group meeting.

-Tomi. Continue work on short term route planning, assist in creating demo for sgm.

-Anu. Try to figure out how to transform screen coordinates to geo coordinates. Work on demo for sgm.

14.3.2012

What have you done since last meeting?

-Jussi. Serious code inspecting since last 50 hours.

-Tomi Connected routewidget and coreupwindscene/route through uwcore. Tried to paint route on map..

-Anu Worked on screen/geo coordinate transformation and demo for sqm.

-Antti Worked on demo for sqm. Progress with getting boat gps coordinates.

Did you have any impediments?

-Jussi. no.

-Tomi. no.

-Anu. no

-Antti. no.

What are planning to do until the next meeting?

-Jussi. Try out something with Toolbox/UI.

-Tomi. Try to figure out why long term route not really working. Continue short term when able.

-Anu. Continue work on demo.

-Antti Translate GPS coordinates to view coordinates.

20.3.2012

What have you done since last meeting?

-Tomi. Been trying to fix long term route planning.

-Jussi. Started work on DB layers manager.

-Antti. Figured out how to use NMEA simulator on Windows. Has been looking for a

way to make coordinate transformations.

-Anu. Debugging long term route planning, same as Tomi.

Did you have any impediments?

-Tomi. Lacking understand of what does the code do in "real world" at times. like coordinate spaces and their effects.

-Jussi. none.

-Anu. Debugging not working partially on QT development environment.

-Antti. none.

What are planning to do until the next meeting?

-Tomi. Keep on trying to find a solution for long term route planning error.

-Jussi. Keep on layers manager work.

-Antti. Translate coordinates.

-Anu. Continue debugging long term route planning.

27.3.2012

What have you done since last meeting?

-Tomi. Changed so that long term now loads geo-coordinates instead of graphic, path calculus still not working.

-Anu. Tried to resolve long term route planning problems.

-Jussi. Keep on layers manager work. Trying to import code from old version.

Did you have any impediments?

-Tomi. no.

-Anu. no.

-Jussi. no.

What are planning to do until the next meeting?

-Tomi. Continue trying resolve long term route planning problems.

-Anu. Continue trying to resolve long term route planning problems.

-Jussi Carry on importing code from old version.

5.4.2012

What have you done since last meeting?

-Jussi. Imported LayersManager and related files from old code to the new code. Fixed the project so that the code compiles.

-Anu. The long term route planning issues fixed.

-Tomi. Worked on resolving the issues with long term route planning.

Did you have any impediments?

-Tomi. Accidentally deleted chart layers from DB and when they were recompiled they were missing columns

-Anu. no.

-Jussi. no.

What are planning to do until the next meeting?

-Jussi. Familiarize myself with OGR libraries. Make changes so that separate chart layers can be loaded independently when selected.

-Anu. Work on chart colors and layout.

-Tomi. Start working on documentation and do some finalizing steps on long term route planning .

10.4.2012

What have you done since last meeting?

-Tomi. Focused on building a document to be used for future upwind project groups.

-Antti. Converting coordinates.

-Anu. Worked on chart colors and layout.

-Jussi. Worked on chart layers.

Did you have any impediments?

-Tomi. no

-Antti. no

-Anu. no.

-Jussi. no.

What are planning to do until the next meeting?

-Tomi. 50 hours left, plan to spend that time with the documentation.

-Antti. Converting coordinates.

-Anu. Continue work with chart colors and layout.

-Jussi. Continue work with chart layers.

PROJECT PLAN FOR UPWIND::SPRING2012

Anu Pramila
Andrei Vainik
Juha-Matti Hurnasti
Tomi Sarni

May 21, 2012

Contents

1	Introduction	1
2	Scope of the project	1
2.1	What has been done so far?	1
2.2	Scope of the UpWind::Spring2012	1
3	Limitations	2
4	Schedule	2
4.1	Meetings	2
4.1.1	Policies	2
4.2	Implementation	3
4.2.1	Development	3
4.2.2	Sprints	3
5	Project deliverables	4
6	Resources	4
6.1	Personnel	4
6.2	Work Environment	4
6.3	Documents	5
7	Risks	5
7.1	Risk analysis	5

1 Introduction

UpWind is an Open Source Software project initiated and coordinated by the UpWind team at Department of Information Processing Science at University of Oulu. Since 2006 there has been several project teams designing and developing an advanced navigation software for sailboats. Today, the software includes all essential navigation features and can be used as such in real boats. After being developed by multiple different teams the software code became difficult to maintain and improve. To improve code maintainability and extendibility a new plugin architecture has been introduced. The plugin architecture allows easy adding and/or replacing of features. Spring 2010 project team designed and reimplemented many of the Upwind navigator software according the new plugin architecture. Fall 2011 team ported most of the remaining software components to the new architecture.

The aim of the Spring 2012 project team is to port the remaining Math-plugin to the architecture. This plugin includes functionalities of automatic long term and short term route planning. The work is done as part of the university course Projekt 2 and each of the project members are expected to allocate 300 hours to the project.

2 Scope of the project

This project continues the work done in UpWind projects in previous years.

2.1 What has been done so far?

The objectives of two earlier UpWind projects were:

- Creating a new, manageable and scalable architecture for the navigator software, with a modular design. This architecture is based on a plugin system.
- Documenting the entire project with UML diagrams.
- Starting implementing the new architecture.
- Port existing code to the new architecture

2.2 Scope of the UpWind::Spring2012

The scope of this project is to port the code of the remaining Math-plugin to the architecture which includes functionalities of automatic long term and short term route planning. Both of these features have been earlier implemented and tested, so the work will be mostly porting of code to the new architecture style. The final part of the project includes wrapping things up and finalizing the new architecture so that the program could be once again used as a whole.

3 Limitations

Project group has four members. Each member has 300 hours to use for the project. This limits the scope of the project as project goals have to be adjusted according to group member's skills and learning curve.

The project has specific goal given by the supervisor Víctor Arroyo, which means the group will follow the project boundaries.

As the work will mostly be done at UpWind laboratory, there is access to some of the sailing boat instruments as well as computers and a server. This limits the work mostly to the laboratory as there is no way to access the server from anywhere else than the laboratory.

4 Schedule

4.1 Meetings

Meeting	Date	Participation	Location
Kick off	23.1	Project team	At university
1st steering group	15.2	Steering group	At university
2nd steering group	14.3	Steering group	At university
3rd steering group	4.4	Steering group	At university
Final steering group	2.5	Steering group	At university
Sprint planning	After each sprint	Project team	At university

The first steering group meeting is for approving the project plan. The Project plan is to be presented to the TOL representative as well as the pre-study report. The second and third steering group meeting's purpose is to verify that the project is on the right track by presenting the achieved results beforehand and receiving feedback about the project's status at the meeting. The final steering group meeting's purpose is to ultimately approve the project's closure and to review the final report.

In the sprint planning meetings the project group will review the accomplishments and failures of the previous sprint with the assistance of the customers representative and discuss what should be done in the next sprint. Tasks of the coming sprint will partially be assigned during the meeting and in the next daily Scrum meetings.

Daily scrums are used whenever the project team is at the same location at the same time. Daily Scrum meetings are short very informal meetings (about 5-15min) to stop and take a look at team status. This means reviewing what has been done since last daily Scrum and what is going to happen next.

4.1.1 Policies

An official invitation will be sent for steering group meetings via email to all project related parties 2 days before the meeting along with related documents.

For less formal meetings an informal email will be sent to remind about the meetings at least one day before the meeting.

Acting as the chairman in steering group meetings will be TOL representative Samuli Saukkonen. After approving the project plan, changes to this document can only be done if all parties involved approve the changes.

This excludes the risks section which can be updated regularly by the project manager.

4.2 Implementation

4.2.1 Development

Project will use the Scrum process model for managing the development.

4.2.2 Sprints

Sprint	Estimated schedule	Main concentration
Sprint 1	23.1 to 10.2	Making the project plan and getting familiar with the working environment.
Sprint 2	13.2 to 7.3	Getting familiar with the code. Locating and identifying pieces of code that are necessary for long term route planning. Some coding done.
Sprint 3	8.3 to 30.4	Coding and testing long term route planning. In the end, long term route planning completed. Preparing for the short term route planning.
Sprint 4	2.5 to 27.5	Long term and short term route planning completed.

Sprint schedules presented above are estimates and they can be renegotiated with Samuli Saukkonen and Víctor Arroyo. However, rescheduling must not affect the final deadline (Final SGM) for the project. According to the estimated sprint schedules, each project member should use around 20-25 hours per week. Contents and goals for each sprint will be decided in pre-sprint (sprint review) meetings. This will be briefly documented and emailed to both Samuli Saukkonen and Víctor Arroyo.

5 Project deliverables

Deliverable	Short description	Delivered to	Delivered at
Project plan	This document	SG	1st SGM
Prestudy report	Research report	TOL	1st SGM
Time management	Working hours	TOL	Before every SGM
Software package	Source codes and binaries of the UpWind application	Samuli Saukkonen	After each sprint
Project portfolio	Project management document	TOL	after the last SGM
Seminar report	Seminar report	TOL	after the project

6 Resources

6.1 Personnel

- Steering Committee
 - Samuli Saukkonen, TOL representative: samuli.saukkonen@oulu.fi
 - Víctor Arroyo, Project coordinator: victor.arroyo@oulu.fi
- Project Group:
 - Anu Pramila, Scrum master/member: AMPramila@gmail.com
 - Andrei Vainik, Project member: antti.vainik@gmail.com
 - Juha-Matti Hurnasti, Project member: jussi.hurnasti@gmail.com
 - Tomi Sarni, Project member: tomi.sarni@gmail.com

The scrum master will be responsible for arranging the meetings involving TOL or the customer. He/she will also be responsible for managing the project-related documents and schedules set in this project plan. Project members are expected to manage their own work and actively participate in the project planning, as well as helping others when needed. Each project member has 300 working hours to use in this project and the work load for one week is around 20 hours.

6.2 Work Environment

The workplace is going to be a room FY1052 at the Department of Information Processing Science at the University of Oulu. The main tools that are going to be used to build the project are:

- Qt 4.7.4: Library for building the application.
- Qt Creator: Software development environment.

- git: Version control system.
- L^AT_EX: Document-making software tool.
- PostgreSQL 9.1.2: Database Manager.
- GDAL 1.8: External library for managing chart data.
- Workstations: computers that are used to design and code the project.
- Operating Systems: Linux, Windows, Os X
- Server: computer to build the project executable file, store the database and version control system.

6.3 Documents

Base document for this project is the project assignment document introduced at the Project II initiation lecture. The document describes the general contents of the assignment and the same information can be found in more detail from this document's Scope of the project chapter. This document describes the project work done at UpWind::Spring2012 by a group of four students that work for the customer under the Department of Information Processing Science of the University of Oulu's supervision.

7 Risks

The risks include technical problems as well as problems among team members. These include the possibility that the lab hardware breaks down or that the project members do not have enough skills to finish the task. All of the risks may end up with delays in the work.

7.1 Risk analysis

1	Technical problems
Description	The lab hardware breaks down
Time	At any time.
Probability	Medium.
Effect	Project crew is unable to work.
Prevention	Computer maintenance and using git.
Threshold	Project crew is unable to work.
Recovery	Contact maintenance, and supervisors .
Notes	None.

2	Lack of skills
Description	Project member does not have enough experience of working in agile mode, C++, Git or OpenGL. Porting of code to new architecture is not familiar concept.
Time	At any time, more likely to be at the beginning.
Probability	Medium.
Effect	Project work is delayed, takes more time to learn to get use to new tools and ways of working.
Prevention	Team members help each other whether in form of mini-courses or informally. Every member continuously studies the tools and methods that are used in the project.
Threshold	Task is not finished in estimated time.
Recovery	Training the team member or reassigning the task for another member.
Notes	None.

3	Code is not working
Description	The code made by multiple project members is not working or compiling.
Time	At any time.
Probability	High.
Effect	Porting the code to support new architecture might fail, having to rollback some changes. In severe cases may affect deadlines and achieving the project plan.
Prevention	Frequent compilations, testing the code by several project members and use the repository wisely.
Threshold	Software does not work.
Recovery	More work needs to be done in finding the source of the problems.
Notes	None.

4	Project member is not able to work
Description	Project member is not able to work in project because of unexpected issues for example illness or some other personal issues.
Time	At any time.
Probability	Low.
Effect	The person is not able to work in a short period. The workload for other members may increase.
Prevention	Open discussion, creating a new <i>shorttime</i> plan for the people concerned by the issue as soon as possible
Threshold	The workload for other members may change.
Recovery	
Notes	None.

5	Underestimation of required work load
Description	Project is too challenging or the scope too broad
Time	At any time
Probability	Low.
Effect	Unfinished system.
Prevention	Choosing appropriate scope for the project. Reviewing plans and possibly rescheduling
Threshold	Remaining work hours are not sufficient to finish the system/task.
Recovery	Reduce scope and team needs to work more efficiently
Notes	None.

6	Sprint targets are not met
Description	The targets set when the sprint was planned are not met in the end of the sprint even though the workload should have been enough
Time	At the end of any sprint.
Probability	Medium.
Effect	Objectives not achieved.
Prevention	Working efficiently and keeping the team motivated.
Threshold	Negative feedback from project evaluations or reviews.
Recovery	Changing the working method. Reviewing the plan and updating if necessary.
Notes	None.

UpWind Project Final Report

Project 2 / Spring 2012

Tomi Sarni, Anu Pramila, Juha-Matti Hurnasti and Andrei Vainik

1. Project Description

UpWind project is software development project with a goal build a sailboat navigation software. The software uses accurate sea charts and can be connected to various instruments such as GPS, wind speed and direction meters and so forth. The software will use the information to plot best possible route option for the sailboat depending on weather conditions and location. The project has been running for several years now and has been subject to multiple student projects. There is a UpWind lab for the project located in room FY1052 in University of Oulu.

Currently UpWind is going through an architectural transformation and our team's purpose was to import route calculations logic from the old version to the new architecture. We were to use Scrum method as a development method. The actual work was done during spring 2012 (February to May).

2. A General Review of the Project Process

Our main task was to import "math" part from the old system which consisted of two main components. The task was divided to be executed in four sprints varying from 3-4 weeks in length. After each sprint the plan was re-evaluated based on the progress we had made so far.

The plan of the Spring 2012 project team was to port the remaining Math-plugin to the architecture. This plugin includes functionalities of automatic long term and short term route planning. The old architecture contained these functions and so according to the plan the work was assumed to be mostly porting of the old code to the new architecture style

Beginning the work and writing the first lines of code was difficult but by the end of the second sprint we had written code towards the long term route planning and the goals seemed achievable. However, in the end of the third sprint we struggled with a coding issue and were forced to reevaluate the goals for the last sprint.

The problems we faced were mostly due to the lack of proper documentation. Therefore, a choice was made to focus solely on long term route planning and use the remainder of the time to make a documentation of the project for the future teams. The documentation now includes guides on how to compile the old and new codes, how to use version control system, where to download required software and what difficulties our team faced and how they were

overcome.

In the end of the project, the long term route planning was completed and documentation readied for the future teams so that they could work more efficiently from the beginning. The long term route planning included calculating the route, painting the route to the screen on the map relative to the zooming level and showing the image of the ship in its proper location. The ship answers to GPS coordinates, wind direction and speed. In addition, some coloring of the maps were corrected to better correspond to official nautical maps.

3. Experiences and perceptions about the software

There are a lot of practical difficulties in starting the project which could have been avoided by having a better documentation or person who we contact when we needed technical assistance and would be available at time we were at the UpWind lab. In a way it was a learning experience being forced set up the development environment yourself but in our opinion that time could have been better spent. Also it is somewhat demotivating not being able to actually tackle the work because of the equipment.

As said in previous chapter we were able to complete long distance route calculation but not the short distance route calculation. Our estimation is that, without a technical supervisor to make design decisions regarding the architecture and to assist the team in understanding the software system more comprehensively, we could have done things better. We did have a person in this technical role but he was most of the time reachable only by email and with a delay and the times he was able to come to the lab was when most of us could not.

We did try to have a good conversation about design decision regarding the software, which we were needed to be done but should have not be done by us.

We feel that the software has progressed during our project.

4. Amount of Work

The Project II guidelines depict that each of the project members have 300h to spend for the project. The realized hours have been collected in the following table. The work consisted

Name	Hours done	
Tomi Sarni	295h / 300h	
Juha-Matti Hurnasti	279h / 300h	
Anu Pramila	289h / 300h	
Andrei Vainik	280h / 300h	

5. Experiences about used methods and tools

The project was managed through Agile methods. Scrum meetings were arranged few times a week and the project was divided into four Sprints. After each sprint, a Steering Group Meeting was prepared and held. Having Scrum meetings was unfamiliar to most of the project team members but meetings were quickly adapted. The Scrum meetings helped in dividing the workload and coordinating efforts towards the common goal.

Coding environment was Qt Creator. Everyone of the project team members were at least somewhat familiar with C++ programming language and therefore applying the Qt Creator was not overly difficult. However, some of the special features of Qt left us puzzled. All of these notions and solutions we found were collected in a document for future teams, so that they need not trouble themselves with the same problems.

The version control was handled by Git version control system. In the beginning of the project we were informed that this system was to be used and we were left with an overly extensive and somewhat aversive documentation. None of the team members were familiar with the git and were thus forced to learn how to use it. Through trial and error the system was put into use but many hours were spent in learning. Had there been guidance or a brief documentation, the process of learning and time used could have been cut tremendously.

6. Experiences about the client

In this project The client was represented by prof. Samuli Saukkonen and Victor Arroyo. The original idea for the project was proposed by prof. Saukkonen as he had noticed lack for navigation software for layman sailors. In this sense Samuli Saukkonen represented user and at the same time an application domain specialist. Victor Arroyo had written his master thesis earlier on this topic, and had been participating the project ever since. Victor represented the technical specialist.

Samuli Saukkonen offered support whenever we requested. Communication was carried out by email, or by visiting the UpWind laboratory whenever possible.

Victor Arroyo was in contact with project group nearly on daily basis. He offered technical support to any technical related problems that group came across. Most of the communication was done over internet chat, because Victor works daily and has a very busy schedule. Most of the visits by Victor were done on his personal time. Sometimes we would have needed more support and internet chat wasn't sufficient mean of communication to discuss highly technical and complex problems, but unfortunately we had to cope with the situation. The high need for face-to-face support was no doubt result of insufficient documentation.

7. Summary of experiences

Anu: The project taught us a lot although I'm not sure if it was about working in a project. There were no big deadlines to hit, nor there were any project funding applications to be sent. University as a workplace gives you some freedom about when you are working and maybe even about what you are supposed to do. However, it comes with a cost of writing applications, publications and tedious bureaucracy. During this project we wrote couple of documents and spent rest of the time coding. I would be thrilled if in my Ph.D. I could spend all my time just working towards my Ph.D instead of shifting papers. In that sense, this project work was not much like a project work (at a university) generally is.

We did have a nice team and we were able to work well together. There were some confusion now and then about what each one of us was doing at the moment and even some overlaps, but they were resolved quickly enough. I feel that a lot of time of the project were spent on studying work that someone else could have explained in five minutes in the beginning of the work. For example, had there been someone from the previous project team to help us begin would have helped tremendously.

Tomi: I felt at times that the whole UpWind project was put in place to teach and simulate how projects work in real IT firms. The tools and methods that we were supposed to use were similar to those being used in real firms. By real firm i mean companies that actually function under financial pressures. Putting four people that do not know each other beforehand and drop them in middle of a 6-7 year legacy software without any real support and give them a set of tasks with deadlines to do resembled a sort of worst case scenario. The learning curve to get into the project felt extremely steep. I think this sort of setting gave strength to follow through the project but it also took away motivation to do anything extra. I made a decision just to do the hours i was expected with best of my abilities and thats it. I am certain that with some different type of a project, i would have engaged the project with more enthusiasm. All in all working on project that might never see the daylight (no

financial pressure) but still try to simulate the difficulty of a poorly handled disaster project ultimately made completion of the project like chewing through bitter chalk.

Beside the negatives i felt that the chosen tools and development environment and having a place to work were the best parts of the project. So even though i had some troubles motivating myself at times to work 110%, i did appreciate the fact that the time i am putting into this project is not time wasted. All in all, i believe this project gave me much better set of skills and readiness to work in a real project in a real IT firm. In comparison to many of the other available projects in the Project 2 - course.

Juha-Matti: First of all, the biggest thing this course taught me was the meaning and necessity of documentation and knowledge management in projects. Since previous project groups had not put so much effort on these issues, it caused us a lot of extra effort to even understand what to do and how to do it. But then again, situations like these probably happen in the real world as well since switching jobs today is quite common practice. Also, working with legacy code is common today since not very often organizations implement a completely new information systems without any legacy systems. So therefore very valuable lessons have been learned on this course. But due to the reasons mentioned, I was not able to learn those things I sought to learn on this course in the first place. My goal in the beginning of this course was to learn more about C++ and especially about Qt, but not much learning happened on those areas. To be honest, this course worked the other way around: at the moment I have no motivation or interest towards learning to use Qt at all. And that probably wasn't the aim of this course.

Andrei: In the beginning I was very enthusiastic about the project, because it sounded like a real world application that had actual meaning, in contrast to usual exercise works done at TOL. The project idea sounded quite technical and as I learned to know the development environment it, sounded even more technical, which was great.

In the beginning I think all of us had problems getting familiar with the project and development environment. The software itself was around 6 years old, and had been developed by numerous project groups. This usually leads to not so practical solutions, when more than one chefs work in the same kitchen. Our job was to isolate certain component of the software from old version and encapsulate it in new version. At some points this task seemed like a nightmare, taking into account that it took me about one week just to compile the project. Also we had to learn to use new tools, which wasn't an

easy task either. The amount of information to handle was quite large. But in my opinion this was the good side about the project, because this is what programming is all about.

During the project I learned new technologies, such as: QT and GIT and gained some experience about working in a team. Our team consisted of four people, with different programming backgrounds and we got along quite well. The client was also quite nice. The only “bad” side was the lack of documentation from previous groups and lack of immediate support. I would recommend this project to future Project II participants.