Project Plan for Upwind::Spring2012

Anu Pramila
Andrei Vainik
Juha-Matti Hurnasti
Tomi Sarni

February 13, 2012

Contents

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 P̈roject 2änd 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 bee 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
3nd 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 used for steering group meetings via email to all project related parties at least 2 weeks 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 Arryo. 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.
- LaTeX: 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 *shorttime* 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 neccessary.
Notes	None.