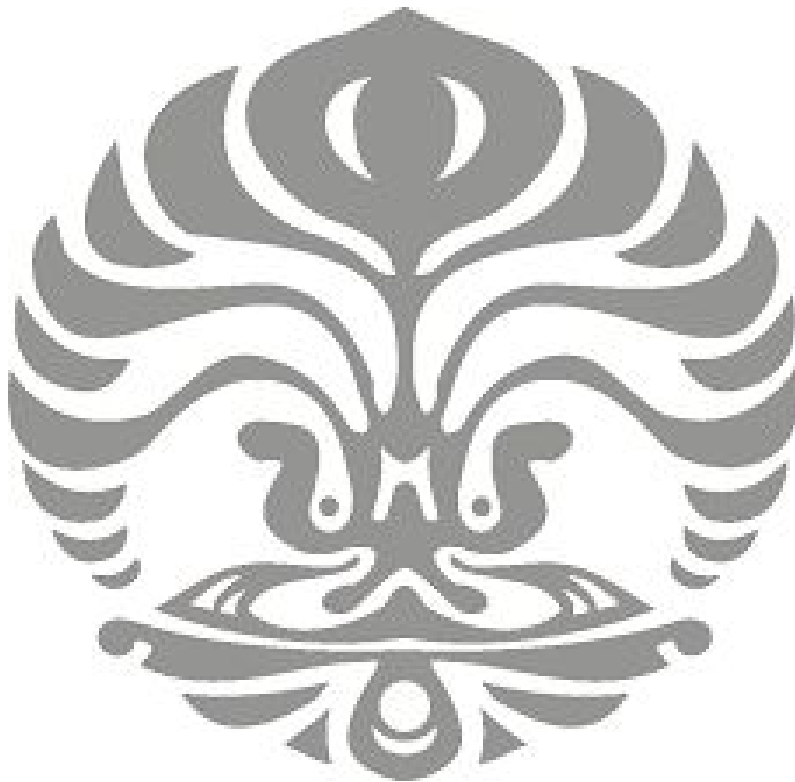# Aware-D : Voice Recognition-based Driving Awareness Detection

Software Project Management Plan

v.1.2

Yudi Andrean Phanama

Cécile Duthoit

Department of Electrical Engineering, Faculty of Engineering

University of Indonesia

Depok, Indonesia

2015

TABLE OF CONTENT

DOCUMENT STATUS SHEET

Version	Release Date	Responsible Party	Major Changes
1.0	October 4th, 2015	Project Manager	Initial Document Release for Comment
1.1	October 5th, 2015	Project Manager	Changes in process model, added explanation
1.2	December 5th, 2015	Software Manager	Changes in team configuration, added explanation

1. Introduction

1.1 Project Overview

In this fast paced world, mobility of people is very important and transportation is a very essential element to support the mobility of people. One of the most used transportation in the world are cars. Cars, or private cars are used by people to travel quickly from a place to another through the roads, which not seldom take a long time in roads and traffic. There are an estimated number of 1.2 Billion cars active in the roads worldwide, and will grow to 2 billion in 2035, showing the importance and demand increase of cars.

Driving a car needs the driver to be in a proper condition and aware of the road to ensure safety. In many countries, driving a car is restricted to rules limiting the activities done in the car while driving, and the condition of the driver. Drivers in a not proper condition (drunk, sleepy, on drugs, etc.) can cause harm to the other traffic user, in form of a potential to incite accidents on the roads. According to the National Highway Traffic Safety Administration (NHTSA) 33,561 people died in traffic crashes in 2012 in the United States (latest figures available), including an estimated 10,322 people who died in drunk driving crashes, accounting for 31% of all traffic deaths that year.

Our project, Aware-D, which tries an approach of human-computer interaction, is made to help the driver and the families (or relatives) of the driver to ensure the driver's condition while driving. With this app, the driver will be provided questions in form of voice, and the driver must answer the questions with voice too. The app will then compare the answers of the driver with the pre-allocated answers in the database of the app to determine if the driver is alert (aware) or not. The app can then notify the selected person the condition of the driver. With Aware-D, we hope that we can help reduce the accidents rate because of loss of awareness problems (from tiredness, alcohol, drugs or any other possible cause), and we hope to contribute to this field of study.

1.2 Project Deliverables

The list of project deliverable is:

- Software Project Management Plan
- Software Requirements Specification
- Software Design Document
- Project Test Design Document
- Project Risk Management Plan (included in SPMP)
- Final Document

1.3 Evolution of the SPMP

This document is a subject to changes. The assumptions, dependencies and constraints for the project, the detailed time and resource planning for each phase can change as the project runs. Changes in this information will create a new SPMP with new version number, but with the same status.

1.4 Reference Materials

Below is a list of references that we use to completion of this project:

1. https://trello.com/
2. http://developer.android.com/index.html
3. http://stackoverflow.com/
4. *Taylor, James: Managing Information Technology Projects - Applying Project Management Strategies to Software, Hardware, and Integration Initiatives, AMACOM, 2004*

1.5 Definitions and Acronyms

Acronym	Definition
ADM	Administrator
PR	Programmer
IM	Iterative Model
DD	Database Designer
PM	Project Manager
RC	Resource
SDD	Software Design Document
SPMP	Software Project Management Plan
SM	Software Manager
SRS	Software Requirement Specification

2. Project Organization

This section specifies the process model for the project and its organizational structure.

2.1 Process Model

The project process will be based on Iterative Development Model, which is effective for dynamic software development, with the possibility to go back to the previous phase to dynamically improve and modify anything.
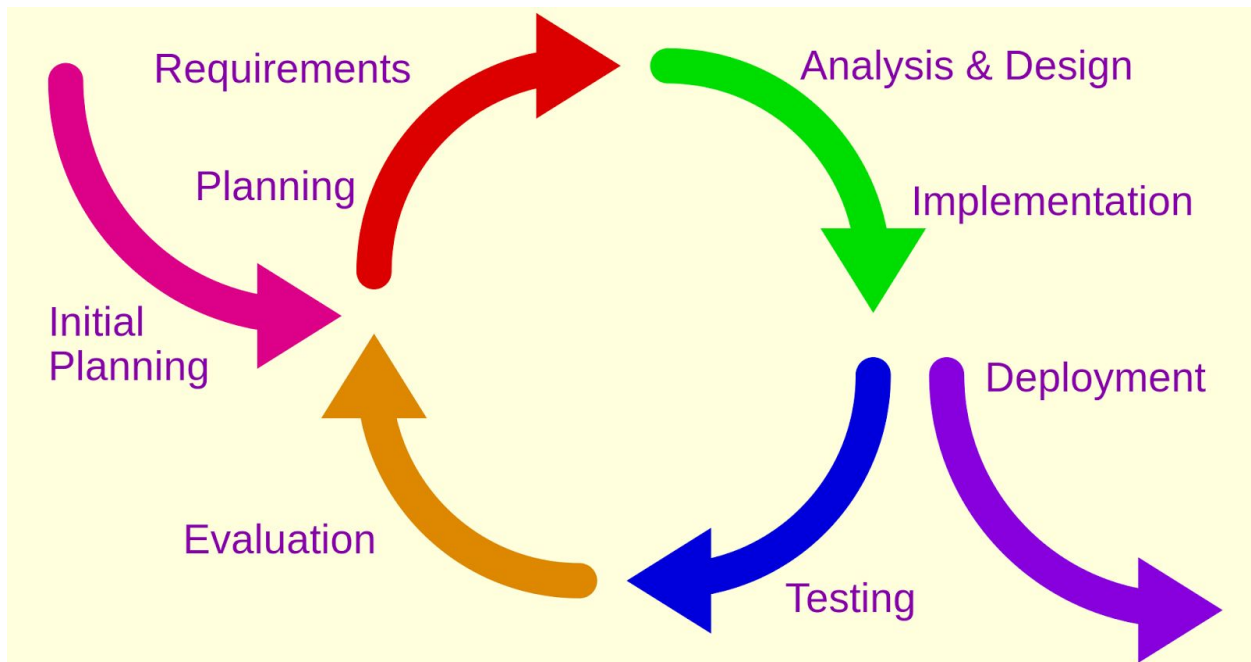
The project is divided in these phases:
1. Inception
2. Elaboration
3. Construction
4. Transition

The inception phase is the initial phase, which consist of initial planning of the project, and the planning of the whole project. Elaboration is the next phase, with the step of requirements analysis and the analysis of the project, with the initial design. The construction phase will be the longest phase in term of time, with the implementation and deployment step in it. The last phase, the transition phase,will be consisted of testing step and evaluation step, in order to validate the product and do the launching, while maintaining the product from the test results.

2.2 Organizational Structure

Role	Name	E-mail	Phone
Project Manager	Yudi A. Phanama	yudiandreanp@gmail.com	+62-89693-959600
Software Manager	Cecile Duthoit	duthoit@etud.insa-toulouse.fr	+62-81281-961369
Administrator	Yudi A. Phanama		
	Cecile Duthoit		
Programmer	Yudi A. Phanama		
	Cecile Duthoit		

2.3 Project Responsibilities

Our team consists of two persons, therefore, we divide our project into several tasks. Every person has his job and to do lists. Here is the task divisions made to complete this project:

Role	Description	Person
Project Manager	1. Planning scope, activity, and resource 2. Scheduling tasks 3. Manage risks and issues 4. Reviewing documents 5. Keeping the team's work 6. Verification and Validation	Yudi Andrean
Software Manager	1. Producing technical specification 2. Leading the programming team 3. Manage the overall code development process 4. Initiate and control the testing 5. Models the architecture of the app	Cecile Duthoit
Administrator	1. Secretary work 2. Create timeline, cost estimation, documents	Yudi Andrean Cecile Duthoit
Programmer(s)	1. Implementation 2. Code documentation 3. Database design 4. Data gathering	Yudi Andrean Cecile Duthoit

3. Managerial Process

3.1 Management Objectives and Priorities

Our team's philosophy is to produce great software while maintaining the utmost level of harmony within the team. All elements like TL, PM and SM have to work hardly to achieve the objectives. TL monitor the work progress of team in every week. The project flexibility matrices can be seen in section below. The goals of our project are 1) to produce the best possible product within the given constraints, 2) help people driving safer

Project Dimension	Fixed	Constrained	Flexible
Cost		X	
Schedule		X	
Scope (functionality)			X

Table F-1: Flexibility Matrix

3.2 Assumptions, Dependencies, and Constraints

Our team's assumption is this software will be run on Android OS 4.0 and later version. This software also will be developed with Android Studio. A successfully tested version must be ready within three month by December 7th, 2015. This project will use resources in the form of time and effort that will be spent developing the project deliverables.

3.3 Risk Management

This section mentions a number of possible risks for the project. Also, actions or measures are described to prevent or to reduce the risks. We only discuss the most important risks.

1. Miscommunication

   Probability: Medium

   Prevention: After a meeting, one of the group member creates an interview report (in Trello). Every participant and every person who should have been a participant of the meeting should get a copy of this report. Team members should not hesitate to ask and re-ask questions if things are unclear.

   Correction: When it becomes clear that miscommunication is causing problems, the team members are gathered in a meeting to clear things up.

   Impact: High

2. Time shortage

   Probability: High

   Prevention: Care is taken to plan enough spare time.

   Correction: When tasks fail to be finished in time or when they are finished earlier than planned, the project planning is adjusted. If time shortage becomes severe, user requirements, which have low priority, are dropped.

   Impact: High

3. Design Errors

   Probability: Medium

   Prevention: The design should be reviewed very critically.

   Correction: When errors in the design are noticed, all the work that depends on the faulty design should be halted until the error is corrected.

   Impact: High

4. Illness or absence of team members

   Probability: Medium

   Prevention: PM should reminded his/her team members to make attend the meeting and also try to find a good timing so that all the member teams can attend the meeting. If someone cannot come to meeting, he/she should contact the team leader.

   Correction: By ensuring that knowledge is shared between team members, work can be taken over quickly by someone else if a person gets ill.

   Impact: Medium

5. Member(s) quits

   Probability: Medium

   Prevention: -

   Correction: Restructure all the documents and staffing

   Impact: Medium

3.4 Monitoring and Controlling Mechanisms

In order to monitor and control this project progress, our team do several things such as:

1. Weekly project group meeting

   The group meeting will be held weekly every Thursday from 05.00 pm until 08.00 pm at the library. Later, due to schedule changes, this weekly meeting has been replaced on every Wednesday from 05.00 pm until 09.00.

2. Back up the project's data

   In order to make share the same knowledge between the team members, all the project's data will be upload to Github so that anybody can have the data. If this option cannot be done because of the technical issue, all the team member should at least keep the latest data about the project.

3.5 Staffing Approach

Skills required for the building of project are:

1. **Technical Skills**

   Able to use some of programming language like Java, PHP, SQL and software development tools.

2. **Management Skills**

   Creating Documents, maintaining Project Members Tasks and managing the project.

3. **Social Skills**

   Able to present to the customer and convince that our projects could be a part of the solution of the loss of awareness while driving.

4. Technical Process

4.1 Methods, Tools, and Techniques

Tools:

- Computing System: Windows 10 (With JRE 8), Intel quad-core i7 2.2GHz and Windows 8.1 (With JRE 8), Intel processor
- Target System: Android 4.0.0 Ice Cream Sandwich and later versions
- Development Environments: Sublime 3, JDK 1.8
- Programming Language: Java (Android framework)
- Software Development Tools: Android Studio (Gradle Based) v1.x

Methods:

- Documentation   : GitHub, Trello
- Programming   : Object-Oriented, Java (Android Studio)

Techniques

- Testing: Black Box Testing, White Box Testing.

4.2 Software Documentation

The software documentation is kept in the same servers and directories as the source code itself. It is managed using the same tools as the source code. We are using GitHub and Trello as a media for documentation. To ensure that the implementation of the software satisfies the requirements, the following documentation is required as a minimum:

1. The document follows the baseline of IEEE 1058-1998 SPMP
2. Documents must be reviewed, after some points of the Projects
3. The purpose of document reviews is to get docs of high quality

4.3 User Documentation

User documentation applied in every program codes. Project uses online Github (for online repository with push and pull system) to maintain project codes and backup files of project progress. The technical implementation on the codes itself will use JavaDoc documentation to implement the documentation of the codes.

4.4 Project Support Functions

Configuration management

- The purpose of software configuration management is to plan, organize, control and coordinate the identification, storage and change of software through development and transfer. It will be done by the Software Manager, he will inform all the members when a change needs to be made.

Verification and Validation

- Software Verification and Validation activities check the software against its specifications. This part will be done by the SM, based on the SRS and SDD.

5. Work Packages, Schedule, and Budget

5.1 Work Packages

Work Package	Role	Hours estimated
Software Project Management Plan	PM	48
Software Verification and Validation Plan	PM	10
Unit Testing Plan	SM	12
Integration Testing Plan	SM	12
Software Testing Plan	SM	12
Acceptance Testing Plan	SM	12
Software Project Management Plan	PM	32
User Requirement Documents	ADM	20
Graphical Design	PR	28
Software Requirement Documents	SM	30
Prototype	SM	120
Database	PR	180
Software Design Document	SM	150
Code	PR	720
Software User Manual	ADM	24
Formal reviews	PM	12
Meetings	PM	28
Presentations	PM	2

5.2 Dependencies

Project management plan must be done first instead the other document or even for the application, because the progress about developing document depend on project management plan. More even, application that we build also depend on software and hardware specification, so we have to decide it first. After that, project verification and validation document plan and

also design test document that we made depend on application that we did before. Final document also can be done after application and system testing finish first.

5.3 Resource Requirements

The most important resources are human resource and the required software and hardware. Other resources may include the sufficient place to do the code, works, and meetings, network connectivity, things that supporting the group's work convenience. The hardware and software requirements are listed below.

Hardware

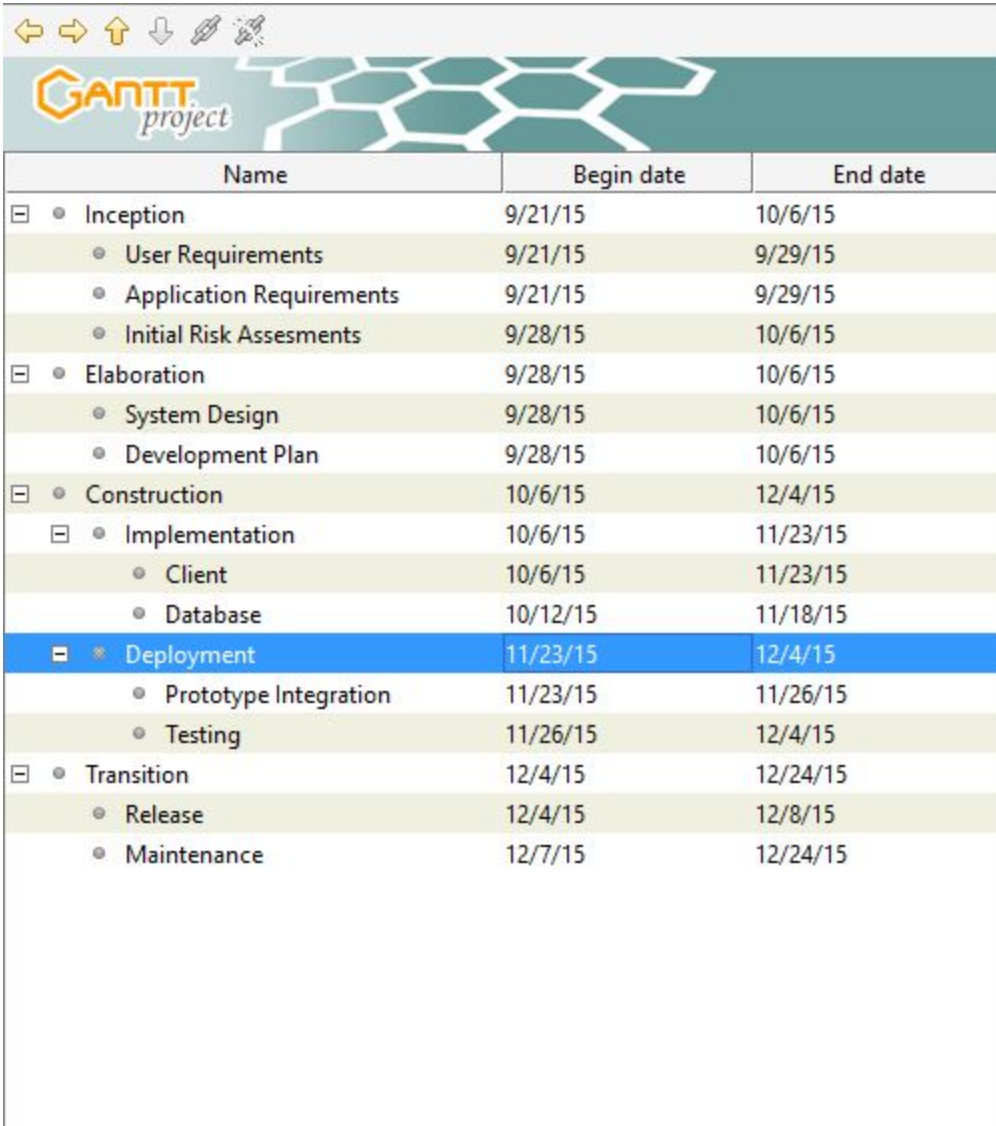- PC with Android Studio and JDK1.8, Smartphone with Android 4.0 version above, and a server

Software

- Github, Trello, Sublime Text 3, and Android Studio

5.4 Budget and Resource Allocation

This part is subject to changes, here are the resources and budget allocations we need. The table below is described about the budget of the hardware that are needed in this project.

Device	Price (Rupiah)
Asus N45SF	7.600.000
Android Device	1.500.000
Internet	200.000

5.5 Schedule

Name	Begin date	End date
⊟ ● Inception	9/21/15	10/6/15
● User Requirements	9/21/15	9/29/15
● Application Requirements	9/21/15	9/29/15
● Initial Risk Assesments	9/28/15	10/6/15
⊟ ● Elaboration	9/28/15	10/6/15
● System Design	9/28/15	10/6/15
● Development Plan	9/28/15	10/6/15
⊟ ● Construction	10/6/15	12/4/15
⊟ ● Implementation	10/6/15	11/23/15
● Client	10/6/15	11/23/15
● Database	10/12/15	11/18/15
⊟ ● Deployment	11/23/15	12/4/15
● Prototype Integration	11/23/15	11/26/15
● Testing	11/26/15	12/4/15
⊟ ● Transition	12/4/15	12/24/15
● Release	12/4/15	12/8/15
● Maintenance	12/7/15	12/24/15

Figure. The Schedule timeline of the project