

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

Software Test Documentation

v.0.3



**Yudi Andrean Phanama**

**Cécile Duthoit**

Department of Electrical Engineering, Faculty of Engineering

University of Indonesia

Depok, Indonesia

2015

# Table of Contents

- 1. Introduction
  - 1.1 Test Objective
- 2. Test plan
- 3. Test design
  - 3.1.1 Speaking module :
  - 3.1.2 Database module :
  - 3.1.3 GPS module :
  - 3.1.4 SMS module :
  - 3.1.5 Graphical User Interface :
- 4. Test cases
  - 4.1 Sets of inputs :
    - 4.1.1 Speaking module :
    - 4.1.2 Database module :
    - 4.1.3 GPS module :
    - 4.1.4 SMS module :
    - 4.1.5 Graphical User Interface :
  - 4.2 Conclusion of the series of tests :
    - 4.2.1 First tests
- 5. Test procedures
  - 5.1.1 Speaking module :
  - 5.1.2 Database module :
  - 5.1.3 GPS module :
  - 5.1.4 SMS module :
  - 5.1.5 Graphical User Interface :

# 1. Introduction

This document is an overview defining our testing strategy for the Aware-D application. Its objective is to communicate project-wide quality standards and procedures. It portrays a snapshot of the project as of the end of the planning phase. This document will address the different standards that will apply to the unit, integration and system testing of the specified application. We will utilize testing criteria under the white box and black box. Throughout the testing process we will be applying the test documentation specifications described in the IEEE Standard 829-1983 for Software Test Documentation.

## 1.1 Test Objective

The objective our test plan is to allow us to find and fix as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal.

We will test our mains methods and check the user-friendship of our user interface.

## 2. Test plan

That part will explain how our tests will be executed. We will proceed to unit tests of each module of our application.

There are 5 modules : the speaking module (divided into the listening module and the speaker module), the database module (divided into the file importation and the database reader), the GPS module, the SMS module, and the Graphical User Interface module (divided into 4 different activities).

For each part of each module, we will expose the input and output expected, with their legal and illegal values and what will be done in case of an illegal value as explained below.

- Legal input values are test values within boundaries of the specification equivalence classes. The inputs should respect these boundaries to make the function successful.
- Illegal input values are test equivalence classes outside the boundaries of the specification. The inputs that do not respect these boundaries will rise an error that should be handled. That error handling will be also explained in the boundaries tables below.

We will precise for each module if the tests are black box tests or white box tests, as defined below.

- Black box testing typically involves running through every possible input to verify that it results in the right outputs using the software as an end-user would. For some functions, that testing method will not be used because there is no contact with the user interface. For example, database tests will be done with white box testing method because the only way to access the inputs and outputs is to manipulate the code.
- In white box testing, inputs and outputs are tested directly at the code level and the results are compared against specifications. This form of testing ignores the function of the program under test and will focus only on its code and the structure of that code.

The tests will be executed by someone outside the conception of the application, in order to take a global view of the app and the bugs that can occur.



### 3. Test design

#### 3.1 Unit tests :

As explained in the Test Plan, we will in that part describe the inputs and outputs of each module of our application, with their legal and illegal values, and the error handling in case of illegal value.

##### 3.1.1 *Speaking module :*

###### 3.1.1.1 *Listening Module :*

- Black box testing :
  - input : user speech
  - output : text representing the right speech

###### INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Internet connection state	ON	OFF	Prompting the user to put it on
Internet connection speed	Reasonably fast	Slower	Ask the user to speak again
Words pronounced	Everything in the answers database	Everything else	False answer If nothing detected : question repeated (up to 3 times, the fourth will be considered as a wrong answer)
Number of words pronounced	1 or 2	0 and more than 2	False answer

###### OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Words understand	Words spoken by the user	Everything else	Display the recognized word(s) to allow the user to check if it is the word(s) he/she pronounced
Output type	String	Everything else	DataTypeError exception

###### 3.1.1.2 *Speaker Module*

- White box testing :
  - input : String text from the database
  - output : speech with good pronunciation of the right text

###### INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Words input	Everything in the database	Everything else	SQLiteException
Data type	String	Everything else	DataTypeError exception

###### OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Speech output	Speech of the input text	Everything else	

Data type	String	Everything else	
-----------	--------	-----------------	--

### 3.1.2 Database module :

#### 3.1.2.1 Importing the file :

- White box testing :
  - input : SQLite format file containing the database
  - output : SQLite format file in the app system

INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
File research	The file must be found	File not found	Message displayed with Log.i ("the file should be in the asset folder and it is unfindable)
File format	SQLite	Everything else	Message displayed with Log.i

#### 3.1.2.2 Database reader :

- White box testing :
  - input : SQLite String query with the given index of the question
  - output : Cursor object containing the String from the database

INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Index	From 1 to MAX_INDEX	< 1 and > MAX_INDEX	ArrayList index exception
Index data type	Integer	Everything else	DataTypeError

OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Type	String from the database matching the query	Everything else	

### 3.1.3 GPS module :

- White box testing :
  - input : position from the GPS
  - output : longitude & latitude

INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
GPS sensor state	ON	OFF	Prompting the user to put it on

#### OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Longitude & latitude valued data type	Double	Everything else	

### 3.1.4 SMS module :

#### ➤ White box testing :

- input : Double value from the GPS module + user's name from the database

*The user's name is already tested in the GUI part, when the user enter it.*

- output : String

#### INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Network state	ON	OFF	Prompting the user to put it on
Latitude & longitude from the GPS module data type	Double	Everything else	DataTypeError
User's name	String with only letters	Any other type and String with numbers or symbols	Prompting the user to enter his/her correct name ( <i>in the GUI part</i> )

#### OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
SMS sending	SMS sent	SMS not sent	Message displayed "SMS sending has failed"
Google Maps link type	String	Everything else	DataTypeError

### 3.1.5 Graphical User Interface :

#### 3.1.5.1 Main menu (first activity) :

#### ➤ Black box testing :

- input : button clicked and user's name
- output : System action

#### INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Start button	Clicked	Unclicked	Wait until it is clicked
User's name data type	String with only letters	Any other data type and String with symbols or numbers	DataTypeError

#### OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
------	-------------	---------------	----------------

Move to the second activity	Moved	Not moved	Display message “Move to the next activity has failed”
-----------------------------	-------	-----------	--

### 3.1.5.2 Question view (second activity) :

- Black box testing :
  - input : Exit button clicked
  - output : next activity (Drive Now or Don’t Drive or back to main menu if Exit clicked)

#### INPUT BOUNDARIES :

If the user wants to exit :

Type	Legal input	Illegal input	Error handling
Exit button	Clicked	Unclicked	Wait until it is clicked

#### OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Move to the next activity	Moved	Not moved	Message displayed “Move to next activity has failed”

### 3.1.5.3 Drive Now view (third activity) :

- Black box testing :
  - input : Drive Now button clicked
  - output : next activity (Question view)

#### INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Drive Now button	Clicked	Unclicked	Wait until it is clicked

#### OUTPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Move to the next activity	Moved	Not moved	Message displayed “Move to next activity has failed”

### 3.1.5.4 Don’t Drive view (fourth activity) :

- Black box testing :
  - input : Back to main menu button clicked
  - output : next activity (Main menu view)

#### INPUT BOUNDARIES :

Type	Legal input	Illegal input	Error handling
Back to main menu button	Clicked	Unclicked	Wait until it is clicked

#### OUTPUT BOUNDARIES :



Type	Legal input	Illegal input	Error handling
Move to the next activity	Moved	Not moved	Message displayed "Move to next activity has failed"

## 4. Test cases

In this part, we will develop our sets of inputs in order to test our modules as explained in the previous part. We will describes the events that occur in outputs and conclude about the efficiency of our application and the improvements we need to add to solve the potential errors.

### 4.1 Sets of inputs :

#### 4.1.1 Speaking module :

##### 4.1.1.1 Listening Module :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
Internet connection state	ON	Legal		
Internet connection state	OFF	Illegal		
Internet connection speed	Reasonably fast	Legal		
Internet connection speed	Very slow	Illegal		
Words pronounced Question : how many letters in the word "cat" ?	"Three" correctly pronounced	Legal		
Words pronounced Question : how many letters in the word "cat" ?	"Three" badly pronounced	Illegal		
Words pronounced Question : how many letters in the word "cat" ?	"Four" correctly pronounced	Illegal		
Words pronounced Question : how many letters in the word "cat" ?	Nothing (silence) once then "Three" correctly pronounced the second time	Illegal then Legal		
Words pronounced Question : how many letters in the word "cat" ?	Nothing (silence) four times	Illegal		
Number of words pronounced Question : what is the capital of USA ?	"Washington" correctly pronounced	Legal		
Number of words pronounced Question : what is the capital of USA ?	"Washington DC" correctly pronounced	Legal		
Number of words pronounced Question : what is the capital of USA ?	"It is Washington" correctly pronounced	Illegal		

#### 4.1.1.2 Speaker Module

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
Data type	Normal database request, i.e. index = 3	Legal		
Data type	Boolean	Illegal		

#### 4.1.2 Database module :

##### 4.1.2.1 Importing the file :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
File research	File correctly put on the asset folder	Legal		
File research	File deleted	Illegal		
File format	SQLite	Legal		
File format	.txt	Illegal		

##### 4.1.2.2 Database reader :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
Index	12	Legal		
Index	253	Illegal		
Index data type	18.2	Illegal		
Index data type	ABC	Illegal		

### 4.1.3 GPS module :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
GPS sensor state	ON	Legal		
GPS sensor state	OFF	Illegal		

### 4.1.4 SMS module :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
Network state	ON	Legal		
Network state	OFF	Illegal		
Latitude & longitude from the GPS module data type	Double	Legal		
Latitude & longitude from the GPS module data type	Float	Illegal		

### 4.1.5 Graphical User Interface :

#### 4.1.5.1 Main menu (first activity) :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
Start button	Clicked	Legal		
Start button	Nothing (unclicked)	Illegal		
User's name	"John Smith"	Legal		
User's name	"Fadilla"	Legal		
User's name	"cloclo81"	Illegal		

#### 4.1.5.2 Question view (second activity) :

SET OF INPUTS : (the user wants to exit)

Type	Value	Legality	Event	Expected behavior (YES/NO)
Exit button	Clicked	Legal		
Exit button	Unclicked	Illegal		

#### 4.1.5.3 Drive Now view (third activity) :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
Drive Now button	Clicked	Legal		
Drive Now button	Unclicked	Illegal		

#### 4.1.5.4 Don't Drive view (fourth activity) :

SET OF INPUTS :

Type	Value	Legality	Event	Expected behavior (YES/NO)
Back to main menu button	Clicked	Legal		
Back to main menu button	Unclicked	Illegal		

### 4.2 Conclusion of the series of tests :

#### 4.2.1 First tests

According to the results, we need to improve that part.....blabla

## 5. Test procedures

That part will explain to the reader how to execute each operation of each test of each module, in order to allow anyone to process those tests.

### 5.1.1 *Speaking module :*

The speaking module tests will be the easiest with the GUI tests because they are black box tests and so, they do not need any manipulation of the code.

#### 5.1.1.1 *Listening Module :*

##### 5.1.1.1.1 **Internet connection state**

The tester will just need to activate or deactivate the Mobile Data or the Wi-Fi on his/her mobile phone before answering the vocal question.

##### 5.1.1.1.2 **Internet connection speed**

The tester will have to use the app in a place where the internet connection is reasonably fast (any place in Jakarta should fit if the Mobile Data are used, or at home with a good Wi-Fi signal for example) and in a place where the connection is really slow (for example using the Wi-Fi of the faculty of Engineering on some days) before answering the vocal question.

##### 5.1.1.1.3 **Words pronounced**

When the question "How many letters in the word "cat" ?" will be asked, the tester will answer as described on the test cases table. He/she will pronounce the word written, correctly (with a good and audible English) or not correctly as demanded, or stay quiet.

##### 5.1.1.1.4 **Number of words pronounced**

The tester will answer the question "What is the capital of USA ?" with the exact answer demanded, with a correct and audible English.

#### 5.1.1.2 *Speaker Module*

##### 5.1.1.2.1 **Data type**

The tester will execute a normal request and observe if the input is well pronounced. Then the user will modify one question in the database and put a Boolean instead of the String item. The tester will observe what happens.

### 5.1.2 *Database module :*

#### 5.1.2.1 *Importing the file :*

##### 5.1.2.1.1 **File research**

The tester will verify that the file is properly in the asset folder or delete it, depending on the demand.

#### **5.1.2.1.2 File format**

The tester will check the format of the file (it must be SQLite) or change it into a .txt file, depending on the demand.

#### **5.1.2.2 Database reader :**

##### **5.1.2.2.1 Index**

The tester will give a certain value to the index in the SQL request, according to the demand.

##### **5.1.2.2.2 Index data type**

The tester will give a certain value to the index in the SQL request, according to the demand.

#### **5.1.3 GPS module :**

##### **5.1.3.1 GPS sensor state**

The tester will activate or deactivate the GPS icon on his/her mobile phone, according to the demand.

#### **5.1.4 SMS module :**

##### **5.1.4.1 Network state**

The tester will activate or deactivate the off-line icon of his/her mobile phone, according to the demand.

##### **5.1.4.2 Latitude & longitude from the GPS module data type**

The tester will modify into the code the data type of the SMS module longitude and latitude input, according to the demand.

#### **5.1.5 Graphical User Interface :**

##### **5.1.5.1 Main menu (first activity) :**

###### **5.1.5.1.1 Start button**

The tester will click or not on the start button when he/she opens the app, according to the demand.

###### **5.1.5.1.2 User's name**

The tester will enter the String written in the table when his/her name is asked by the app.

##### **5.1.5.2 Question view (second activity) :**

###### **5.1.5.2.1 Exit button**

The tester will click on the exit button of the question view, or do nothing, according to the demand.

##### **5.1.5.3 Drive Now view (third activity) :**

#### **5.1.5.3.1 Drive Now button**

The tester will click on the Drive Now button of the question view, or do nothing, according to the demand.

#### *5.1.5.4 Don't Drive view (fourth activity) :*

#### **5.1.5.4.1 Don't Drive button**

The tester will click on the Don't Drive button of the question view, or do nothing, according to the demand.

## 6. Quality survey

### 6.1 The form

	Questions	Completely disagree	Disagree	Neutral	Agree	Completely agree
1	The questions are fluently asked					
2	The app is recognizing your exact sayings					
3	The user interface is user-friendly					
4	The app does not crash too frequently					
5	The questions are relevant to detect losses of awareness					

The tester will answer the following questions by choosing between the five (5) possibilities. These questions are about the tester's experience about Aware-D, and should be answered honestly.

### 6.2 Analysis

#### 6.2.1 Responses

By using all the samples of responses from the different testers, we can analyze their experience about Aware-D.

#### CHARTS

#### 6.2.2 What to conclude

We should improve blablabla