Project Plan

Esiri Igbako Amal Kakaiya Simon Jouet

20 January 2011

Contents

1 Introduction

1.1 Identification

This is the Project Plan document for the Linux Embedded Automotive Dashboard (LEAD) team. It contains the plan and the schedules for different tasks and who would be the leader of each tasks. It is susceptible to change as the project requirements change or unforeseen circumstances occur.

1.2 The Team

The Team that was assigned this Project consists of:

- Simon Jouet
- Amal Kakaiya
- Esiri Igbako

Dingkun Ren has defected from the team due to other responsibilities.

2 Team Responsibilities

We decided to assign ourselves some preliminary roles within the team. These roles are to ensure that the team is coordinating and project work runs as smoothly as possible. The roles within the team are stated below:

2.1 Manager

The role of manager was assigned to Amal Kakaiya. As the manager his task is to coordinate the project and make sure it is moving at a pace that would lead to its timely completion. He is also responsible for liaising with the client on behalf the team. This could be for requirements gathering or for validation of the teams work. Authority

2.2 Quality Assuror

The role of quality assuror was assigned to Esiri Igbako. The Quality assuror is in charge of testing that our product meets requirements and that bugs in software are kept minimal. He will be in charge during our testing phase of development. He would also be responsible for editing and proof-reading documents before they are submitted.

2.3 Librarian

The role of librarian was assigned to Simon Jouet. The Librarian is in charge of ensuring that all documentation is kept organised. Also he would be responsible for managing and keeping track of the changes that are made in the documentation and codebase.

3 Authority

The team has allocated a Manager role to Amal, so in this context ultimately the decisions will lie with the manager. However, we will deliberate on any matters that arise as a team so that an informed decision can be made.

Where there are conflicts regarding the final decision to take then the Manager would have the final say. This may introduce a problem whereby some members of the team feel unsatisfied with a final decision. However, we believe we will be able to discuss any conflicts constructively and come to an agreement without this being necessary.

4 Communication

4.1 Meeting Times

As a start, we have chosen to meet in the project laboratory in the Rankine building every Monday and Wednesday from 2:00pm to 4:45pm. These will be fixed meetings times. As the project begins to gather momentum then the team would have to schedule more time to work on the project. This would be discussed when necessary.

4.2 Ad-Hoc communication

The team members have all taken note of each others contact details, mobile phone numbers, and email addresses. We would also frequently us instant messaging to communicate. Also, the trac management system allows us to communicate and track the progress of the project as well.

4.3 Communication with client

Interactions with the client is carried out through our allocated Moodle forum. However we are presently unable to post to the forum. This problem has been raised on several occasions to the project coordinator but we are still unable to post to the forum. The teams alternate solution at the moment is use the email of the client or to try and organise meeting with the client.

5 Documentation and Software Change Management Plan

5.1 Information Management

The team decided to have a dedicated subversion repository hosted on Simons personal server to manage our source code revisions. We will be using the trac management system to manage our project. This trac management system would help to allocate milestones and track bugs. The team would also be using the trac Wiki to share documents and specifications.

5.2 Information Access

Our subversion, and trac system can be accessed anywhere with an internet connection. Therefore, our team will have access to the codebase anywhere they have access to the internet. Changes can be committed by any team member. At every modification of the documentation or source code an email will be sent to the project members to keep them informed with the progress.

6 Task Allocation

Tasks have been broken down to ensure that every member of the team is kept reasonably busy throughout the duration of the project. The tasks were assigned according to preference of each individual in the team.

Predevelopment Tasks

- Meeting with client to discuss specification
- Component Selection
- Research on components
- Procurement of components

The team has just concluded the requirements gathering stage. However further interaction with the client may be needed to verify the validity of the requirements set. The team has also concluded the component selection phase and have made efforts to place orders for the desired components. We would need to wait until the required components have been delivered before hardware fabrication and testing can be done. In the meantime, the software would be developed and tested to see that it fulfils the requirements of our client.

7 Work Breakdown Structure

Hardware Tasks:

- Circuit design
- PCB design
- Module Testing
- Power Supply
- Component soldering and Hardware fabrication
- Hardware testing
- Weather-proof packaging design

Software Tasks:

- Drivers development
  - Socket driver for CAN
  - Keyboard driver using pushbuttons
- HSC08 code development
- GUI Prototyping
- X11 GUI development
- Interface implementation between GUI and driver

8 Perth Chart Illustration

Task	Symbol	Days	Overseer
Circuit design	T1	7	Simon
GUI Prototyping X11 GUI dev	T2 T3	1 7	Esiri Esiri
PCB design	T4	4	Amal
Module Testing	T5	2	Simon
Power Supply	T6	2	Amal
Component Soldering	T7	2	Amal
Fabrication Of PCB	T8	3	Simon
Socket driver dev	T9	3	Amal
Keyboard driver dev	T10	4	Esiri
GUI and driver interface impl	T11	6	Esiri
Weather-proof packaging design	T12	2	Amal

NB: A working day has been realistically estimated to be 5 hours.

Although testing has not been mentioned as a task, extensive testing would be carried out at the end of each of task. The time resource allocated to a task would invariably include time for testing when the task is completed. The logic behind splitting the tasks to software and hardware is to allow both to run simultaneously, thus ending in a uniformed fashion.