Tracie Report

Team 22 *

November 25, 2014

*Members: Mitchell Kember, Do Gyun Kim, Charles Bai, Xiao-Yang (Michael) Min, Min Suk Kim, Renato Zveibil, Leong Si

Table of Contents

1 Summary

“Tracie” utilizes the Scribbler Bot and the Fluke2 chip to trace out drawable inputs given by the user interface, via a web app, through linear movement and precise rotations. Programmed in JavaScript, the client, which runs in the browser, is the ultimate form of communication between the user and the robot. The client provides the user the ability to start and stop the program immediately, and allows the user to modify the data values of the Controller, which alters the distance and time conversion and rotation to time conversion for optimization purposes. The user interface, integrated within the client, allows the user to input a set of points and provides backwards and clearing functionality.

When the program runs, the client communicates with the Server and the Controller, both runs in the terminal, to start up the Tracie script and retrieves information influencing the robot’s performance, such as the speed. The Tracie program is coded in Python using the Myro library, a Python-written framework used for programming robots. The Tracie program supplies the artificial intelligence and program logic to manipulate the Scribbler Bot’s movement.

This report contains the planning diagrams (Work Breakdown Structure, PERT Network Diagram, and Gantt Chart), which provides a strong foundation in optimizing the time spent on each activity through an organized schedule. Additionally, this report includes a requirements document, which identifies the personnel in the group and their respective roles and contributions, and lists the functional and non-functional requirements in order to progress and complete the project. Furthermore, a computation decision making (CDM) section will also be included. CDM is essentially the description of how the problems in the project are being analyzed, as well as the number and description of the criteria and alternatives found. The project retrospective section will include what worked well and what did not in regards to the supplied resources, (such as the Scribbler Bot or Fluke2 chip), team organization, and the development process.

2 Requirements

2.1 Roles

Mitchell Kember(Leader, ...): Set up everything and organize the group. Did 90% of the work.

3 Computation Decision Making

After reading the proposals, we came to a conclusion that we should utilize the drawing feature of the scribbler bot. We decided to create a program which allows a user to input a drawing from a computer. The drawing would then be drawn by the scribbler bot on a piece of paper.

There were many criteria we needed to consider before choosing our final drawing method:

1. Accuracy
   The drawing created by the robot must follow very closely to the drawing on the screen.

2. Time

   We wanted the robot to take as less time as possible to trace out the drawing.

3. Feasability

   Some of our members had little to no experience in programming, especially in Python. We wanted to share our workload equally, so the project should not be too difficulty for anyone.

4. Flexibility

   Our robot would create different drawings based on the user's choice.

With the criteria listed above, we discussed about the possible methods and came up with 3 alternatives:

1) Fixed Shapes

   Allow the users to choose shapes from the Controller. After they have selected their shape, the robot will then draw it.

2) Pixel Based

   The users will create a pixel based drawing on the Controller's user interface, then the robot will recreate the drawing on a piece of paper.

3) Point to Point

   The users will insert “points” on the user interface. Each point is connected directly to the point placed before it. The robot will then follow the points in order, thus create a drawing.

Table 1: CDM of Tracie

Criteria	w_i (%)	Fixed Shapes			Pixel Based			Point to Point		
		C_{i1}	r_{i1}	S_{i1}	C_{i2}	r_{i2}	S_{i2}	C_{i3}	r_{i3}	S_{i3}
Accuracy	30	10.0	1.0	0.30	9.0	0.9	0.27	8.0	0.8	0.24
Time	10	10.0	1.0	0.10	3.0	0.3	0.03	7.0	0.7	0.07
Feasability	30	10.0	1.0	0.30	5.0	0.5	0.15	9.0	0.9	0.27
Flexibility	30	1.0	0.1	0.03	10.0	1.0	0.30	9.0	0.9	0.27
				0.73			0.75			0.85

The result showed that the Point to Point method received the highest point. It satisfied majority of the criteria, therefore became our choice for the project.

4 Diagrams

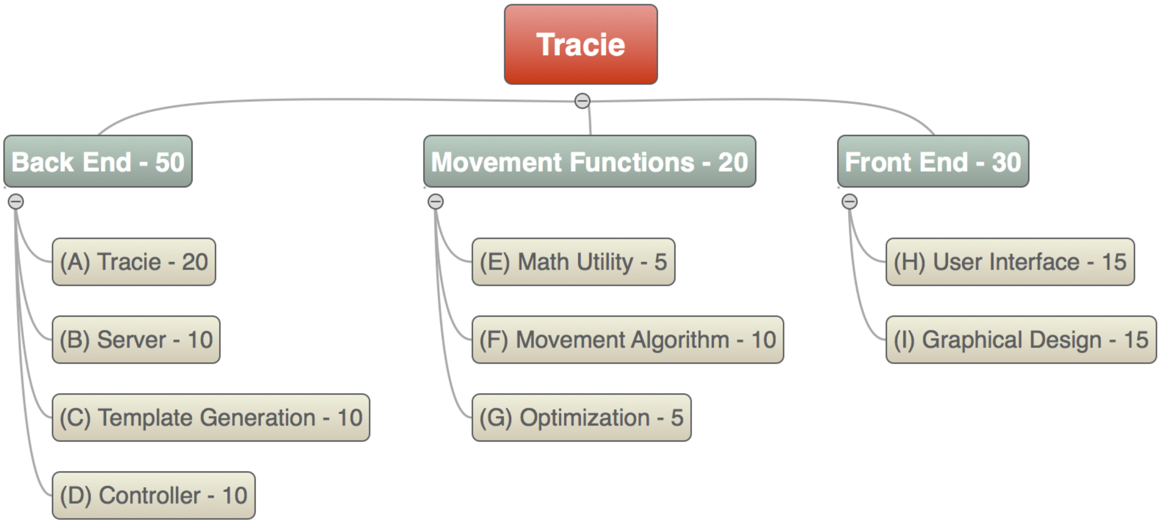

Figure 1: Work Breakdown Structure of Project: Tracie

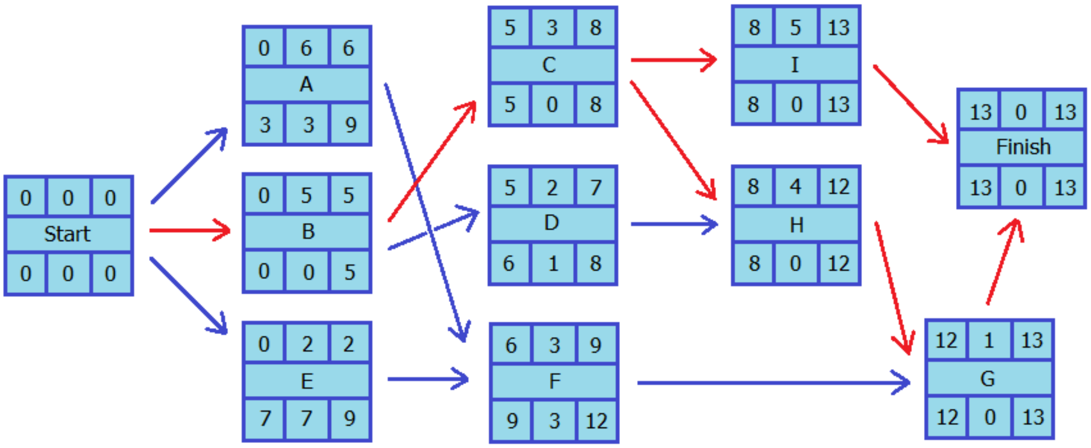

Figure 2: PERT diagram. Critical paths: {B,C,H,G}, {B,C,I}.

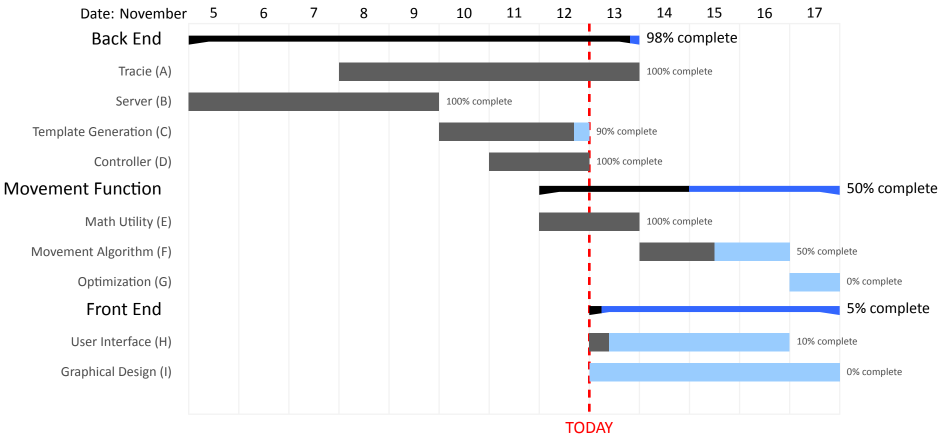

Figure 3: Project: Tracie (Gantt Chart)

5 References