101 Solutions

Iteration 1

101 Solutions

July 26, 2013

Contents

1 Introduction

The purpose of this document is to keep track of the ongoing development progress of the software application manager AppMan. In each iteration additional content will be added which will result in this document being an up to date and relaible method to view progress and current development plans and ideas. Content will include milestones reached, current goals and the current functionality of the software.

2 Milestones reached

- Successfully tendered and accepted to complete this project
- Completed vision and scope document
- Completed an arcitechtural design and documentation
- Completed a prototype GUI for AppMan

3 Immediate Goals

Our next two main goals for the project:

3.1 Server & Client network communication

- Goal: To create a reliable socket connection between the server and client applications, which will serve as a pathway between them.

3.2 Build transference

- Goal: To establish a list of builds which will be recognized by the server application and the ability to copy the builds to the clients(Slave computers)

3.3 Build Information Storage

- Goal: Creating a database that can store information related to builds kept on the master and slaves

3.4 Slave Application

- Goal: Creating a program that will communicate with the Master application

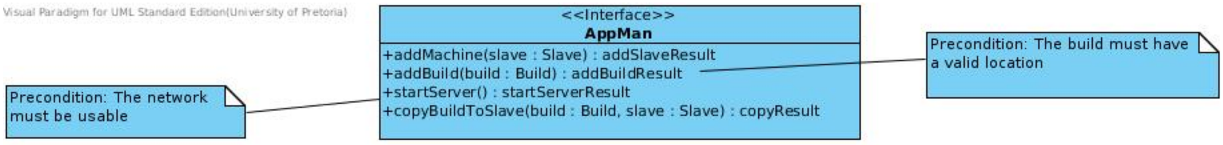
4 Functionality

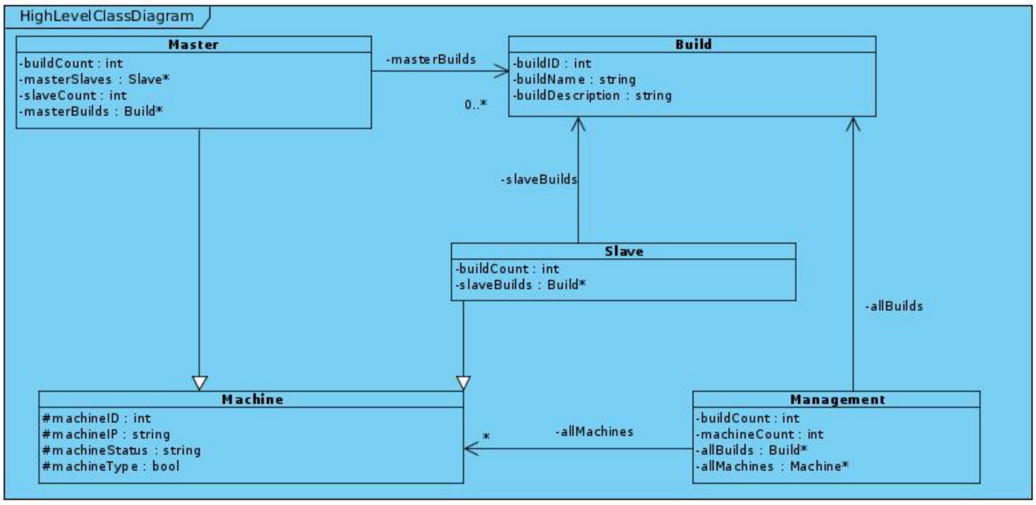
4.1 Graphic User Interface

- A gui built in c++ and QT and deployable in both windows and linux.
- Easy to understand gui layout used
- A help form that will describe some of the workings of the program. This form will display how to add a build, and understand how the program works, etc
- Drag and drop capability on the gui, which will be incorporated with further development
  - Dragging the "drag" label to copy a build from master to slave(future development)
  - Dropping a file or directory on the "+ Add" Button(future development)
- Gui is capable running with or without the images that are to be deployed with it
- Gui Caters for other development
  - Copy files over from master to slave(with a progressbar indicating the file copy progress)
  - Running applications that have been copied to slave computers
  - SpinBoxes that can allow a large amount of slave computers or builds to be shown by the program

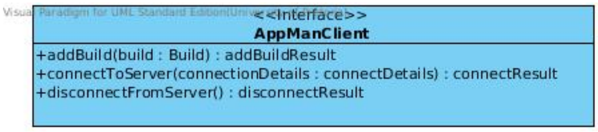
5 System Design

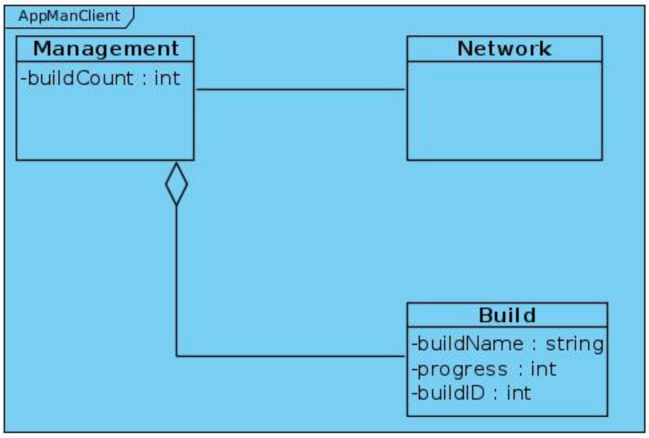
5.1 AppMan


Visual Paradigm for UML Standard Edition(University of Pretoria)
<<Interface>>
AppMan
+addMachine(slave : Slave) : addSlaveResult
+addBuild(build : Build) : addBuildResult
+startServer() : startServerResult
+copyBuildToSlave(build : Build, slave : Slave) : copyResult
Precondition: The network must be usable
Precondition: The build must have a valid location



5.2 AppManClient


Visual Paradigm for UML Standard Edition(Unive
<<Interface>>
AppManClient
+addBuild(build : Build) : addBuildResult
+connectToServer(connectionDetails : connectDetails) : connectResult
+disconnectFromServer() : disconnectResult



6 Processes

6.1 ActivityDiagram

7 Functional Requirements

7.1 Use Case Diagrams

- AppMan
- AppManClient

7.2 Builds

- Builds must be addible to the master
- Builds must be updated on the slave computers once a change have been made
- Only changed files must be sent to the slave computers once a change has been made

- Easy to manage the builds that are on the master computer
- Builds must be able to have the following
  - Name
  - Directory

7.3 Copying

- Copying must be done over the network
- The copy of multiple files must be compressed to reduce the time it takes to copy a file
- Progress of copy must be shown on master computer while copying takes place

7.4 User Interface

- The user interface must be easy to use
- The user interface must make use of drag and drop capabilities to promote usability

7.5 Slave Computers

- Slave computers must connect to the master computer via the network

8 Glossary

- Build - An application build version that could potentially be distributed to slave computers
- Slave - A computer that will be controlled via a master computer
- Application builds will be sent to this computer
- Master - A computer that will control Slaves across a network
- Server - A machine waiting on the network for connections from other machines
- GUI - Graphical User Interface with which a user can control the project

- Project - This project. The distributed application manager
- Application Configuration - Environment variables that are specified when running an application