

Project Planning Mouse Trap

Authors:

Cristina Pêra (up201907321);
Diogo Ferreira (up201805258);
Telmo Ribeiro (up201805124);
Rogério Rocha (up201805123).

TABLE OF CONTENTS



01

Project
Description



02

Requirement
analysis



03

Actor Model



04

Specification



05

Hardware
Architecture



06

Software
Architecture



07

Software
Modeling



08

Gantt Chart

Project Description

The project aims to develop a smart mouse trap system capable of effectively capturing mice while providing real-time notifications and control options to the user via a smartphone application



Requirement Analysis

02

Requirement analysis

- Detect when something is inside the box within 2 seconds.
- Send a notification to the trap owner's smartphone within 3 seconds of detection.
- Allow the user to remotely close or open the box via the smartphone application.

- Provide real-time status updates of the trap (closed/opened) on the smartphone.
- Enable the user to request a picture of the contents inside the box through the smartphone app.

- The warning sent to the owner's smartphone should have a latency of less than 3 seconds.
- The action to close or open the box via the smartphone app should take effect within 3 seconds.
- The system should support multiple mouse traps controlled by the same user/smartphone.
- The trap should aim for practicality through reduced weight/size.
- The Android application should be intuitive to use.



Actor Model

03

Actor Model



Mouse Trap Agent

Capable of detecting events and fulfilling requests

CommCenter

Acts as a broker between the Mouse Trap and User agents

User Agent

The system that provides feedback to the user and accepts controls from them, should also express events and generate requests

Specification

04

Specification

The specification defines roles for Raspberry Pi and Arduino in the mouse trap system. Raspberry Pi acts as the CommCenter Agent, managing communication and multimedia control. Arduino serves as Mouse Trap Agent, controlling hardware components. The Mobile Phone functions as User Agent, interacting with the system through a GUI.

Event Map:

Se - change of sensor states;

Pe - camera information collected;

Oe - motor state = open;

Ce - motor state = close;

Request Map:

Pr - request for camera information collection;

Or - request to change the motor state to open;

Cr - request to change the motor state to close;

Additional:

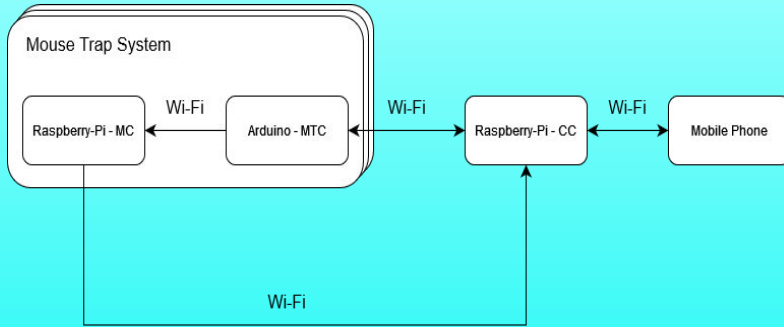
Pc - camera information.

Hardware Architecture

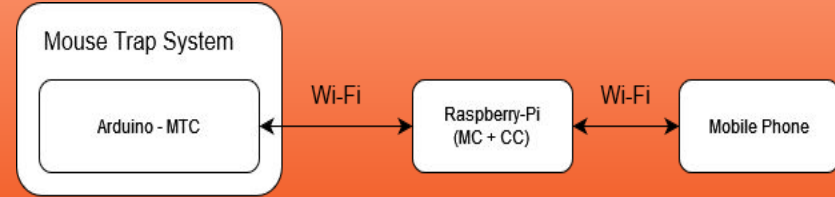
05

Hardware Architecture

Initial hardware architecture diagram



Current hardware architecture diagram

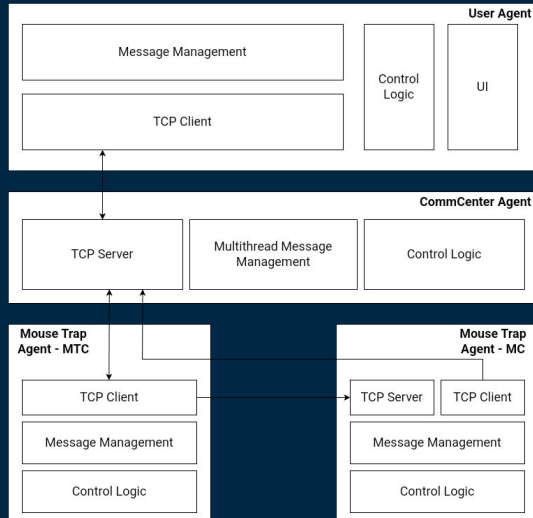


Software Architecture

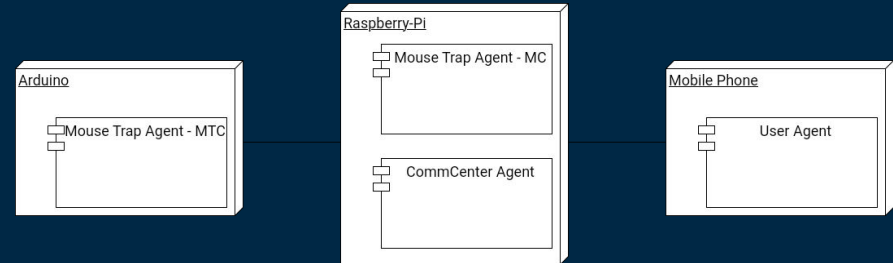
06

Software Architecture

Software architecture diagram



Deployment diagram



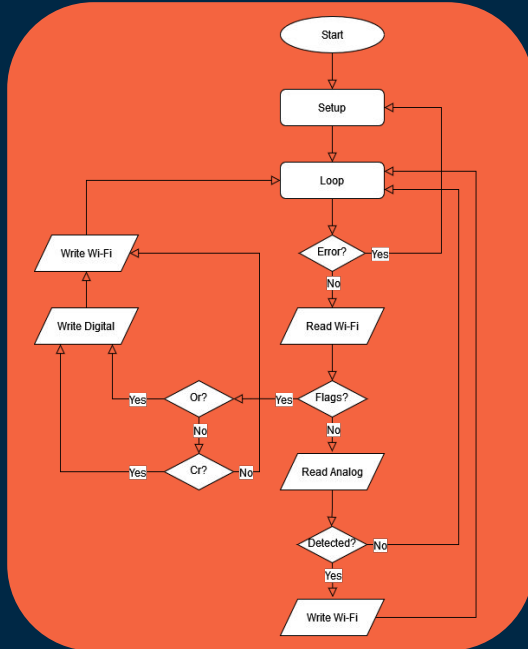
Software Modelling

07

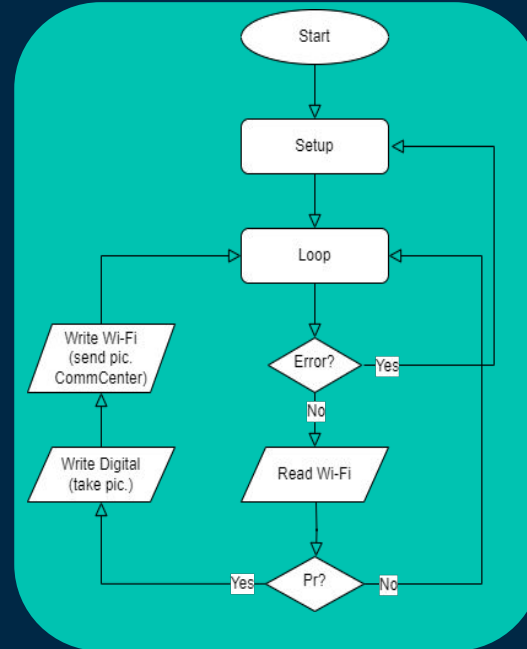
Software Modelling

Flowcharts Mouse Trap:

Mouse Trap Agent – Mouse Trap Controller

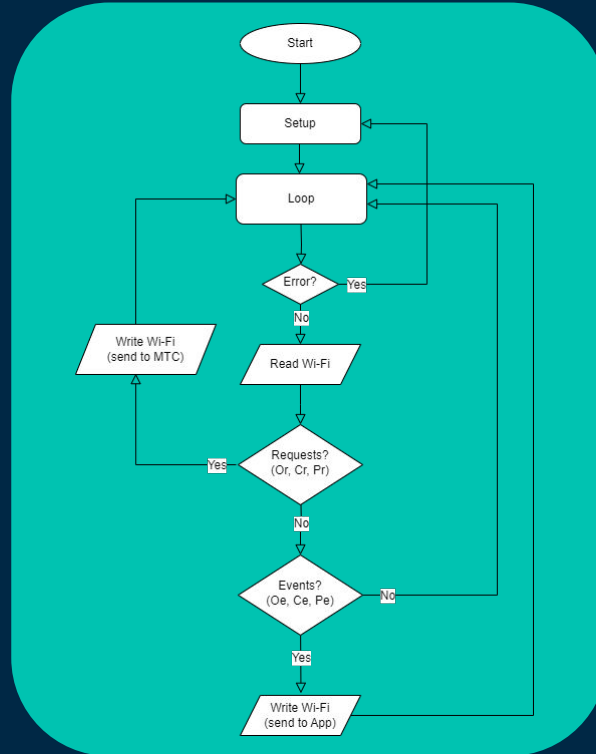


Mouse Trap Agent – Multimedia Controller



Software Modelling (Cont.)

Flowchart CommCenter



Gantt Chart

08

Gantt Chart

