

Hot Potatoes

Learn How To Code

Architectural Document

Group Name:
Hot Potatoes Inc.

Contributors:
Allant Gomez
Chris Mnich
Shahab Shekari
Steven Rengifo
Zachary Guadagno

Rutgers University - Department of Computer Science
CS 431 Software Engineering
Instructor: Alex Borgida
October 15, 2014

Introduction

This document includes some details of the Hot potatoes apis. What this document tries to describe is the Hot potatoes simulator, which helps kids learn how to program using a robot that is picking up potatoes and trying to get “home”.

In today’s world, learning how to program is a very vital skill that many people aren’t taught or exposed to in school. There are many reasons for this, the most frequent one being that learning how to program can be hard, especially for kids. There are many aspects of programming such as syntax and logic that are very hard for children to grasp and the process of learning these things can be very boring. That’s where Hot Potatoes comes in. Hot Potatoes is a game designed to teach kids how to program through a series of worlds in which a hungry robot must collect a series of hot potatoes scattered throughout a world and go home to eat them. Each world will be designed to teach kids about different aspects of programming and to learn the logic of programming in a fun way. The way the game works is the student will program the robot to navigate through the world using a series of commands. Hot Potatoes also allows users to create their own worlds so they can challenge themselves and friends. Teachers can also create an account on the system that will allow them to track the progress of their students and create their own worlds as well, making Hot Potatoes a very valuable tool that makes programming fun and easy.

Structure

We’ve decided to go for an MVC (Model View Controller) architecture to address the structure of our apis. Our Javadoc and UML diagrams are both attached in this archive.

Acknowledgements

Allant Gomez: Javadoc, CodeController, CodeView, Code, CodeBlock, CodeBlock, CodeType, CodeBlockView, CodeView

Chris Mnich: Javadoc, GameController, All Screens & Dashboards

Shahab Shekari: Javadoc, UML, UserController, RobotController, User, Robot, DBController

Steven Rengifo: Javadoc, ActionPanelController, BuildPanelController, MenuController, ActionPanelView, BuildPanelView, MacoPanelView, MenuView

Zachary Guadagno: Javadoc, Block, BlockState, Coordinate, Grid, GridView, GridController,