

Sweet Switches

Educational aspects

Existing approaches to motivate learning of computer programming through entertainment are usually too focused on coding.

Sweet Switches's goal is to promote the development of skills required for solving computational problems. Useful for tasks with:

- Operation order
- Conditional execution
- Repetitions

The trial and error aspect of the game requires the players to:

1. Follow the execution steps
2. Understand where are the pieces causing undesired behavior
3. Fix the problems.

Similar to code debugging

Tangential learning

Besides logical thinking, players may be able to learn about:

Geography:
Brazilian cities on the map

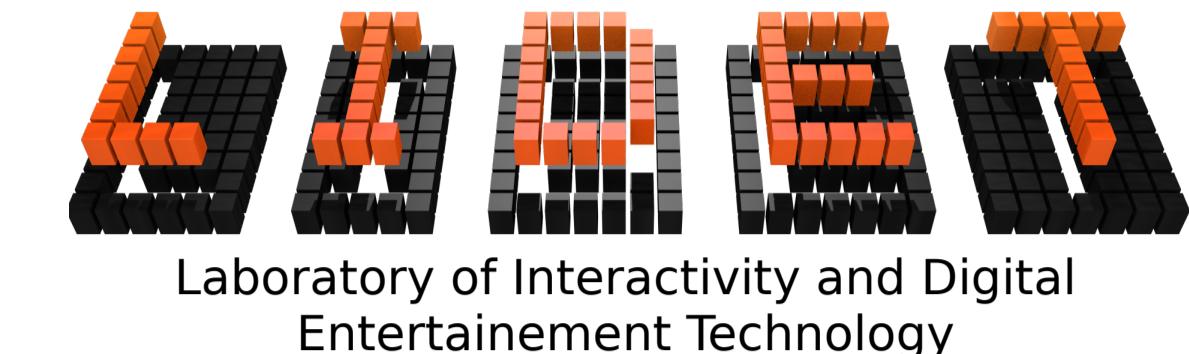
Biology:
Animals of the Brazilian fauna (penguins, jaguars, macaws, etc)



In Sweet Switches players will face a series of puzzles inside ice cream factories. The objective is to attach devices to conveyor belts in order to respond to a production request. As those devices are analogous to programming statements, the game fosters computational thinking.

Vinícius K. Daros, Luiz C. Vieira, Adalberto B. C. Pereira, Flávio S. C. da Silva
Laboratory of Interactivity and Digital Entertainment Technology (LIDET)
University of São Paulo - Brazil
{vkdaros, lvieira, bosco, fcs}@ime.usp.br

Art:
Danilo Gabriel Rios and Salvador Oliva Junior
Art:
Danilo Gabriel Rios and Salvador Oliva Junior
Universidade de São Paulo



Laboratory of Interactivity and Digital Entertainment Technology

Motivation

A report prepared for the UK Computing Research Committee^a advocates that if on one hand learning how to use computers can be seen as similar to learning how to read, on the other hand learning how to program is similar to learning how to write. Both are skills that everyone should have, even though a minority will become professionals (writers or programmers).

^aComputing at School: the state of the nation, 2009

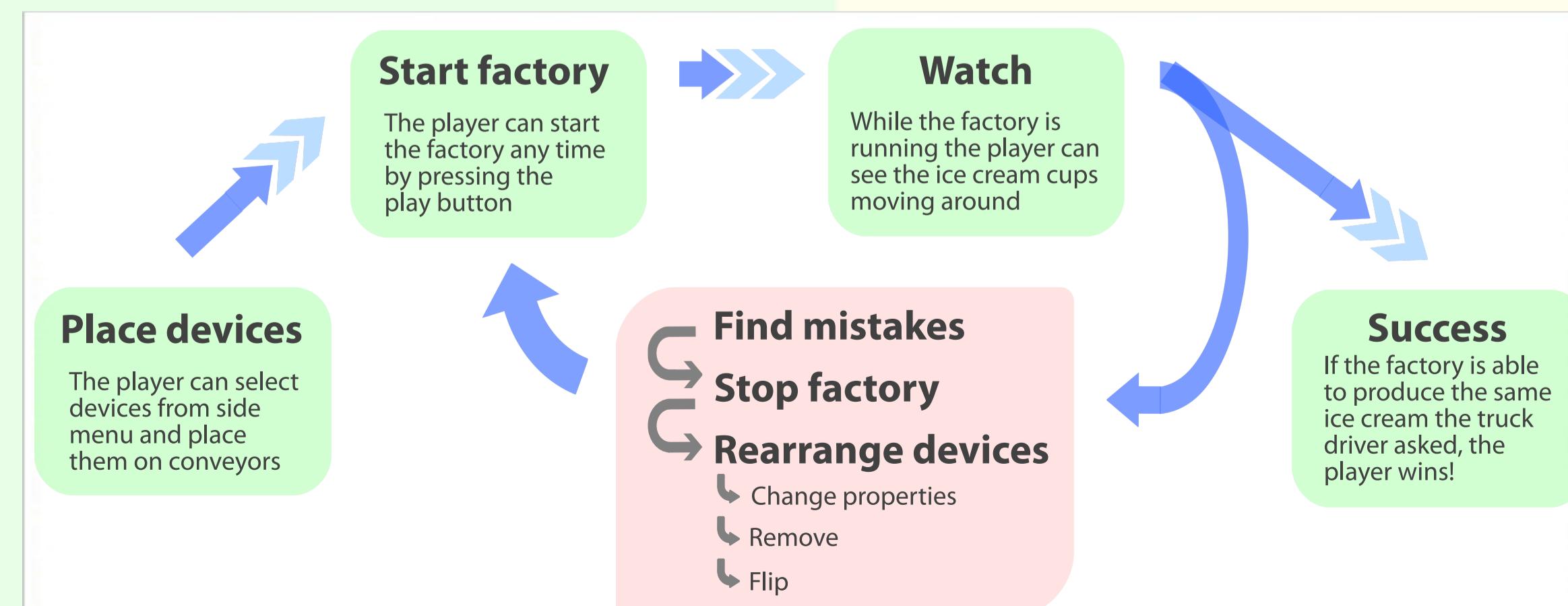
Screenshots

Game elements



Game elements are analogies for programming elements: conveyor segments work as commands, switches work as decisions, and scales work as more complex decisions that allow repetitions (iterations). Starting and stopping the factory line are analogies for testing a program to verify if the sequence of commands and decisions produce the desired result.

Flow of actions



Prototype

Open source project intended for:

- Playability evaluation
- Feedback on characters and visual identity
- Expose at iGam4ER 2014

Development tools used:

Programming language: Haxe

Framework: Flixel

Levels: Tiled

Art: Gimp, Photoshop

Targets

Audience: 12+ years old

Platforms: Web, Android, iOS, Linux, Windows

Find more

