Date: September 8, 2018
To: Manager
From: Zach Walsh, Suzanne Becker
Subject: Programming language for super-hot project

We have decided to use Javascript for this super-hot Santorini project. Our Javascript project will run smoothly on a Linux desktop in the Node.js environment, which is already installed on the CCIS Linux VDIs, as well as on the college's physical machines.

Node provides support for Unix-style standard-in and standard-out I/O operations, as well as for TCP/IP sockets. Additionally, Javascript has native functionality equipped to produce, parse, and manipulate JSON. Altogether, this support will allow us to test, run, and connect distributed project components in a language-agnostic manner.

Javascript allows for the separation of code into folders and modules to give the project codebase a logical structure, enabling us to build robust, dependency-free components that can be reused and tested separately. This will allow us to quickly prototype components in isolation. It can also load code dynamically, so programs will be able to detect when libraries or modules did not load successfully. Our project will be designed to handle such failures gracefully.

The Node package manager provides a diverse array of free, open-source libraries for developing unit test suites and a GUI. Some mature and well-supported unit testing libraries include Jasmine, Mocha, and Chai. We will choose a suitable library for developing unit test suites and tracking test coverage. Although Javascript is often thought of as a front-end web development tool, GUI libraries such as Electron and AppJS enable development of desktop GUI applications using typical web content creation tools like HTML and CSS.

Most modern IDEs support Javascript development to some degree, and Atom in particular has a wide variety of plugins designed to streamline Javascript development. There are plugins for syntax highlighting and function/keyword autocomplete, and it has native Git integration.

Javascript's features and supported Node packages make it flexible enough to satisfy the project requirements and work interoperably with other systems in a distributed fashion. We are excited to make this project a success!