**Software Architecture Specification**

March 21, 2018

Team 2 (Dimitrios Chavouzis, Claire Collver, Clarissa Fung, Drew Hager, Blake Skelton)

1. System and Architectural Context
   1. Rationale

We expect our overall architecture to follow the repository variation of the shared data style. We chose this style since our users will be both accessing and updating our database. We are also considering a client-server architecture style. Since with this style clients are prevented from communicating with other clients and are limited to making requests or receiving replies.

1. Scope

This document covers the architecture guiding the functionality of the entire project, including the website and database.

The project stakeholders include Davidson College (both a user and client), the students of Davidson College (users of the system), and the members of Team 2. Under this scope, we consider ourselves (Team 2), the owners, developers, and testers of the project.

The stakeholders are concerned about the validity of their reservations. Therefore, there is a slight concern regarding the security of the shared data style architecture, since all data connectors are two-way read-write interfaces.

1. Definitions, Acronyms, and Abbreviations

HTTP – Hypertext Transfer Protocol

JDBC – Java Database Connectivity

SQL – Structured Query Language

C&C – Components and connectors

1. Behavior

Our data repository is our MySQL database. Our web system will handle user requests with *data accessors* (JDBC) in order to update the shared data in the database. We also have an administrator component that sets up and monitors the repository (database).



1. Architecture Views
   1. Views

C&C Diagram:

1. Element Catalog

C&C Elements:

* Client
  + Browser that the user is operating on.
  + Allows user to interact with the database indirectly through the server.
* Server
  + The Davidson College server.
  + Connects to our MySQL database.
* Database
  + MySQL database that stores/updates occupancy status as well as holds all the information about the rooms.
* HTTP
  + Protocol that describes how messages are sent back and forth between the client and the server
* JDBC
  + Allows for queries on the database to be made from the web client
  + Connects server to the database.