***Team #***

***Project Name***

Technical Design Document

*To use this template:*

1. *Replace any blue italicized text with your own text. You may remove or add sections as needed for your particular projects.*
2. *Delete these instructions and any other italicized instructions.*

# 1. Tech Stack

# *(The SW development environment)*

*This section will specify your high level technology stack. What languages will you use for development? What databases? What type of source code control? What libraries, tools, IDEs? Are you assuming a specific version of any of those things? Is there a specific processor power / RAM / Disk space required to run the app? Capture all of that information here. Break them into separate sections, using as many as appropriate*

# 2. The Computer Environment

# *(A map of the solution space)*

*Provide details on what is in software, what is in hardware, and what comes from and goes to the environment. Specify the division between clients and servers (if any), and/or between computers (if any).*

# 3. Data Sources, Models, Timing

*No serious application exists without data to drive it. This section will define where the data comes from, what it looks like, and what timing exists for imports and exports, how long the data lives and what happens to it afterwards. Specify temporary vs. permanent (non-volitile) storage.*

## 3.1 Data Sources

*What are your data sources? If there are files that are imported, if your data set has been seeded from a separate source, if administrators need to configure initial data, if the database will be all user entered data - capture that in this section. Where will the data come from? Who will create and maintain it?*

## 3.2 Data Models and Structure

*This section should contain the tables and columns in your database (for SQL) or object definition (for noSQL), and define any JSON, XML, delimited format or other interchange structure that will be derived from those data structures. This should contain detailed information on the purpose, size, allowed values, etc for each data element. It is likely to be a fairly lengthy section, and you should break it up as appropriate. It should also contain an entity relationship diagram (ERD) showing how the data objects / tables interact.*

## 3.3 Timing

*This section should capture timing and method of imports and exports, if any. It should also capture how long the data lives in the system (forever? Last 6 months? Last 2 TB?) and what happens to it when it is removed (Deleted? Archived? Summarized?)*

# 4 System Architecture Diagram

*This should contain a graphical representation of your system, including all machines, databases, user interfaces, APIs, etc. It should show how they are related to each other, and what data flows between them. Everything in the previous sections should have a place here. ALL BLOCKS (representing modules ) AND ARROWS (representing data and status) SHOULD BE LABELED. Hint: this is where most of your grade comes from.*

# 5. User screens

*(sometimes called wireframes)*

*Every menu that a human will access.*

# 6 Deliverables

*What will you be delivering? How will your system be put into production? If you are turning it over to a sponsor, how will that turnover happen? What data will be turned over with it, or will need to be recreated or seeded?*