Software Design Document

for

PlanIT

Version 1.0 approved

Prepared by Christopher Callan, Danny Do, Zach Goad, Timothy Rine, Oseremen Okojie

Louisiana State University

November 4, 2017

1. Introduction

The System Design Document (SDD) describes how the functional and nonfunctional requirements recorded in the Requirements Document transform into more technical system design specifications from which the system is built. The SDD documents the high-level system design and the low-level detailed design specifications.

The SDD describes design goals and considerations, provides a high-level overview of the system architecture, and describes the data design associated with the system, as well as the human-machine interface and operational scenarios. The high-level system design is further decomposed into low-level detailed design specifications for each system component, including hardware, internal communications, software, system integrity controls, and external interfaces.

1.1 Purpose

The Software Design Document documents and tracks the necessary information required to effectively define architecture and system design in order to give the development team guidance on the architecture of the system to be developed. Design documents are incrementally and iteratively produced during the system development life cycle, based on the particular circumstances of the information technology (IT) project and the system development methodology used for developing the system. Its intended audience is the project manager, project team, and development team. Some portions of this document, such as the user interface (UI), may be shared with the client/user, and other individuals whose input/approval into the UI is needed.

2. Architectural Strategies

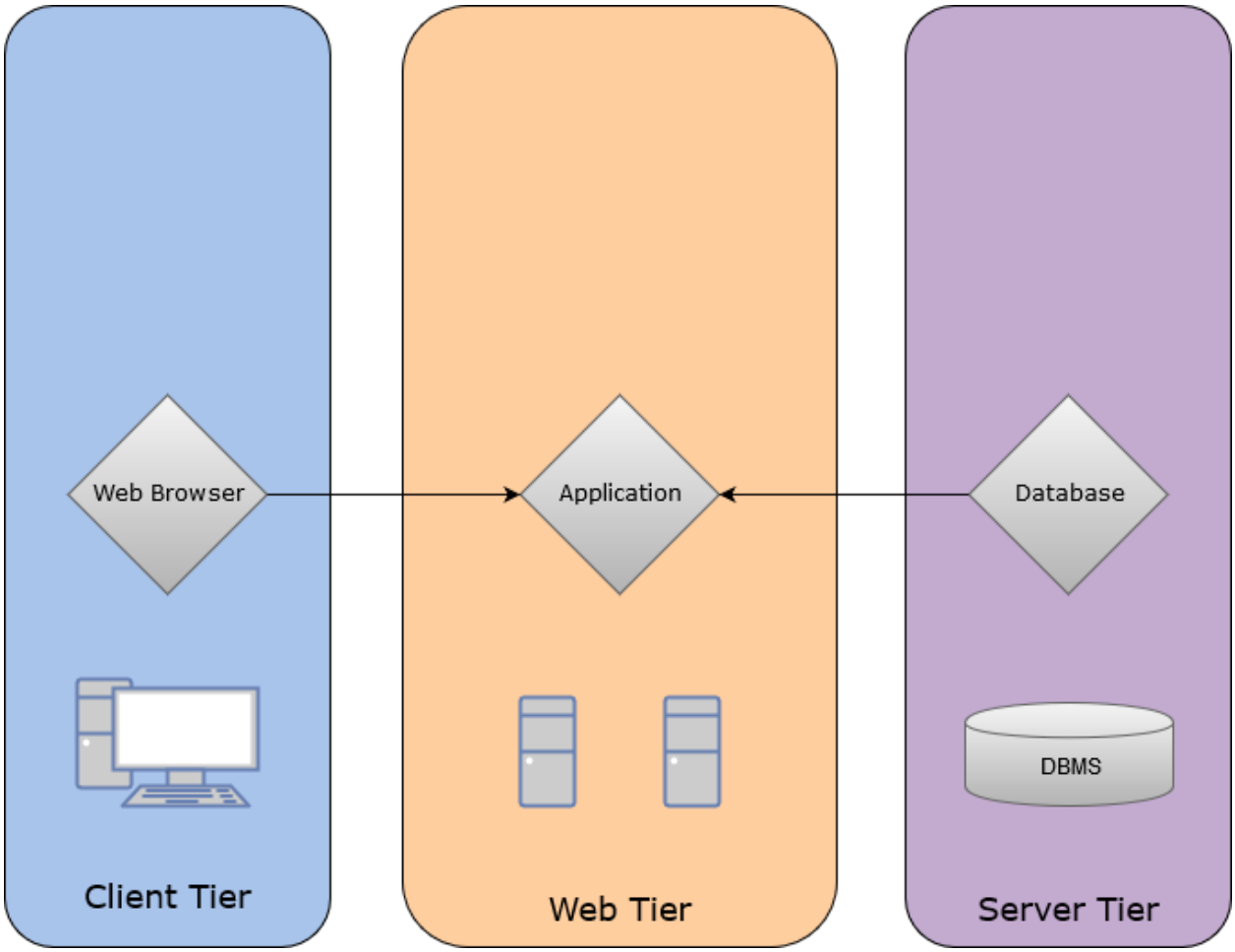
The PlanIt system design has been created using a three-tiered system which is relatively standard for creating small web-based applications. The first tier is the user's PC (or phone/tablet) that connects to the front-end, the second tier contains the web-based application and its functions, and the third tier contains the database & server on the back-end.

These three main tiers are then divided further into subsystems. The Client tier is broken down into the User and Authentication subsystems. The Web tier is broken down into API Management and the application and its features. The Server tier contains the web server and the database that contains all of the relevant data to the main application.

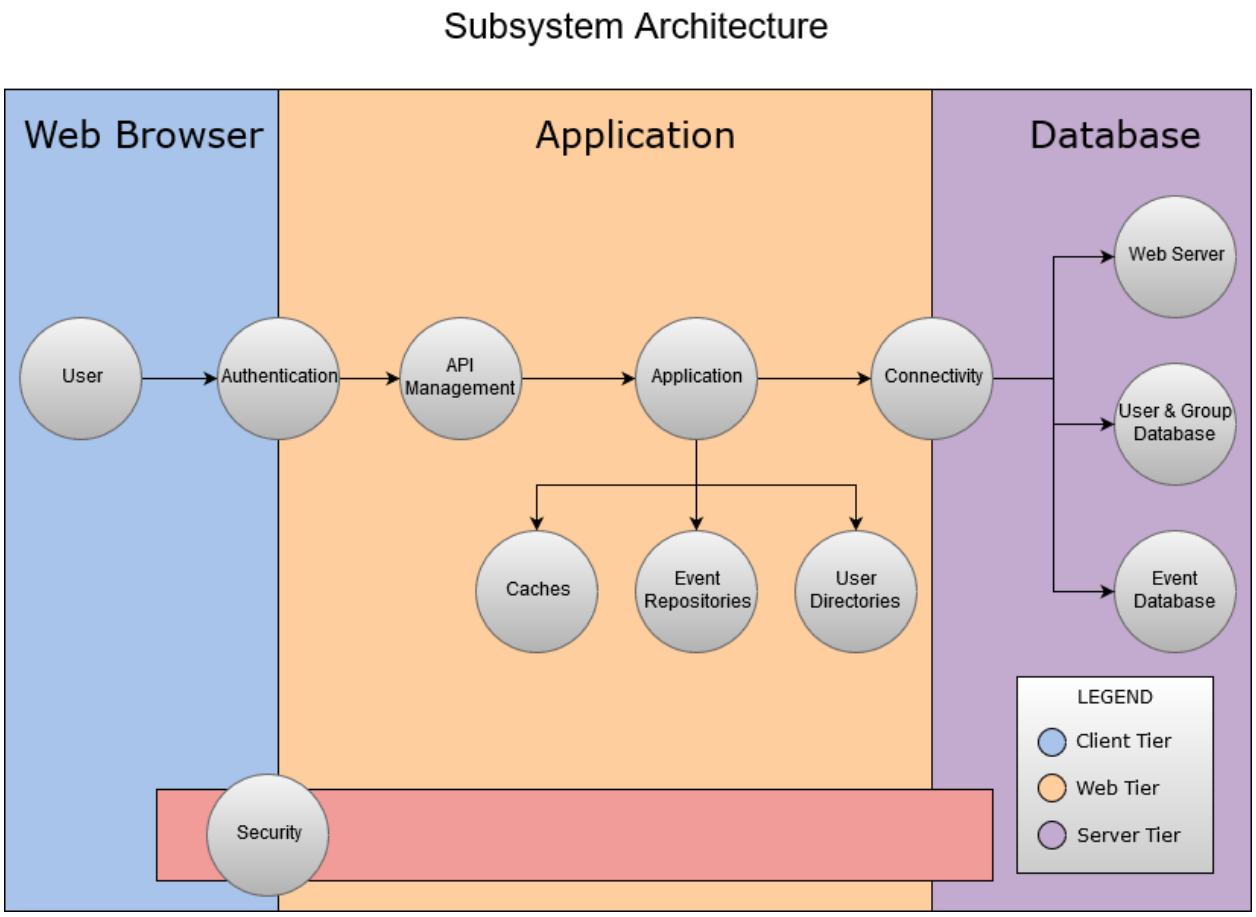
The Web Tier's major design considerations include an easy to use interface, cost-effective development, accessibility on a wide range of devices, and adaptability to growth & requirements changes. For these reasons we decided to develop PlanIt as a three-tiered web-application using JavaScript, HTML5, CSS, and pHp.

The Database subsystem contains the database that stores all of the information needed to run our web application using JSP, JDBC, and mySQL components. We chose to use JSP, JDBC, and mySQL for the database design because those technologies required the least amount of research and learning time, both of which we are limited in.

2.1 Main System Architecture



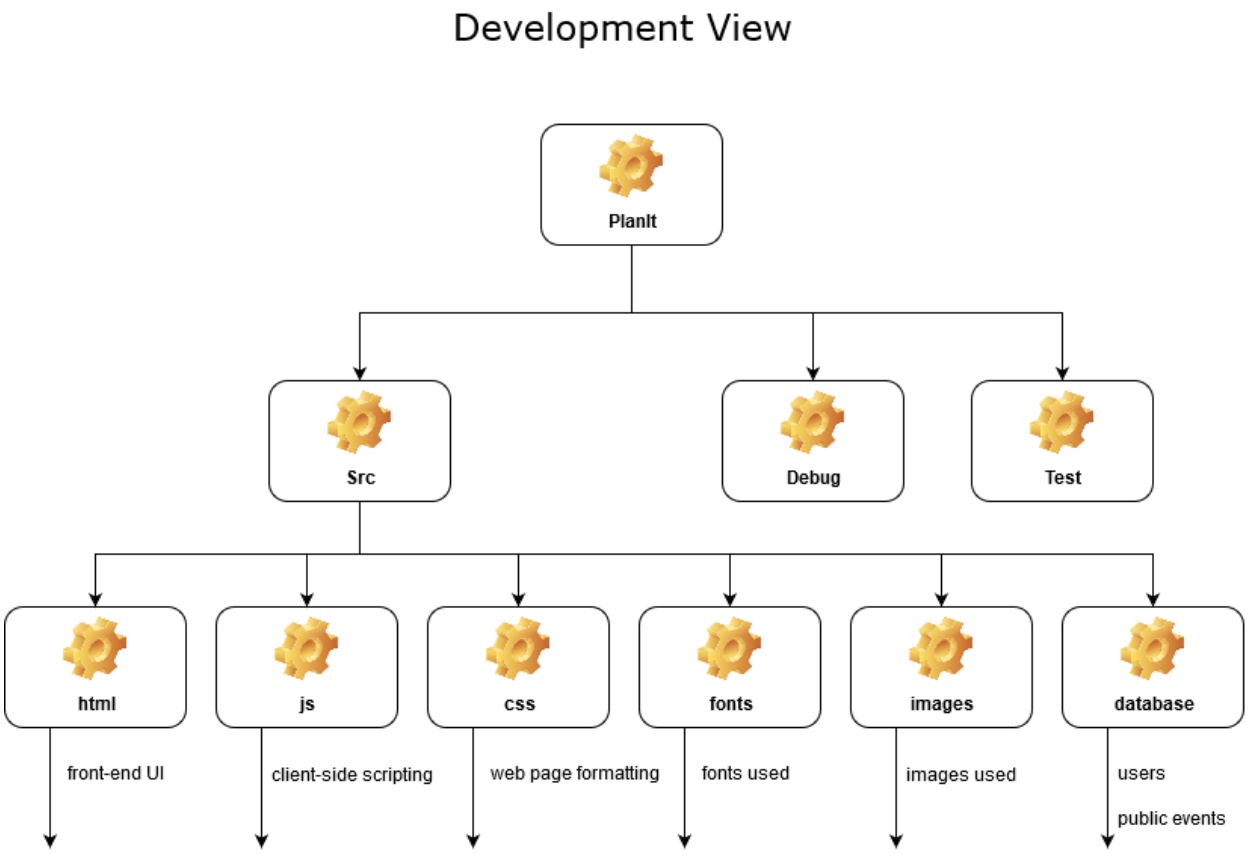
2.2 Sub-System Architecture



3. Development View

The development view illustrates a system from a programmer's perspective and is concerned with software management.

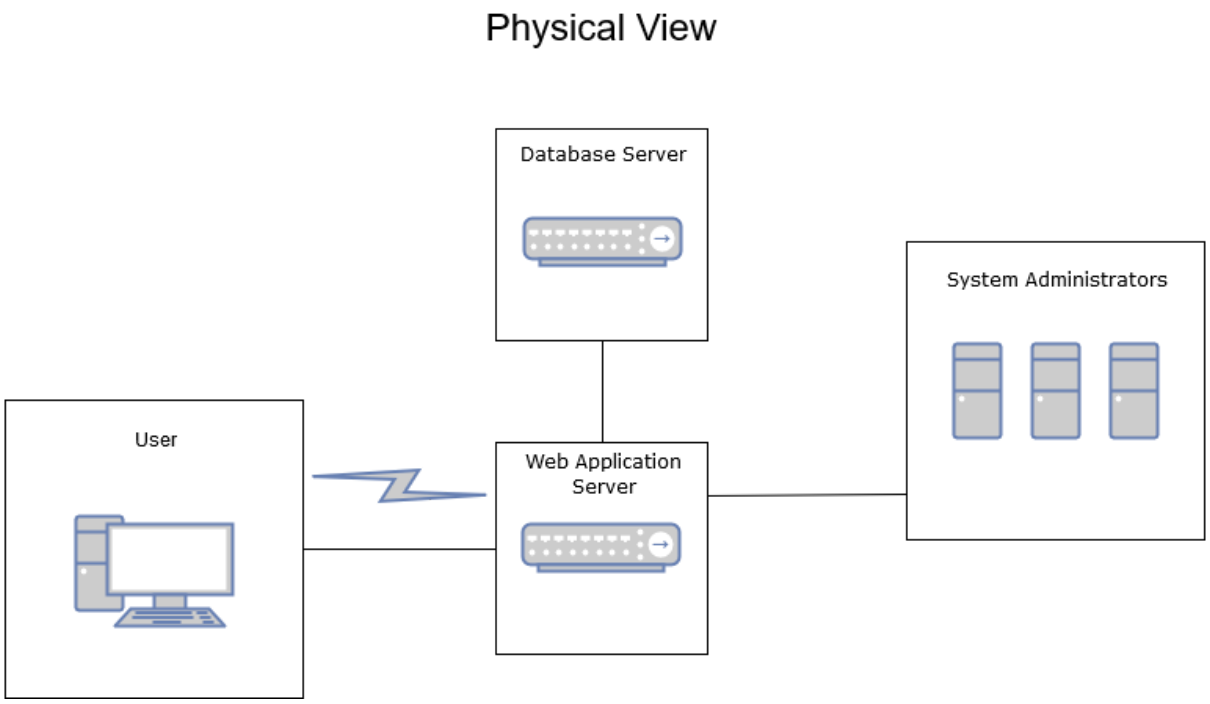
3.1



4. Physical View

The physical view depicts the system from a system engineer's point of view. It is concerned with the topology of software components on the physical layer as well as the physical connections between these components.

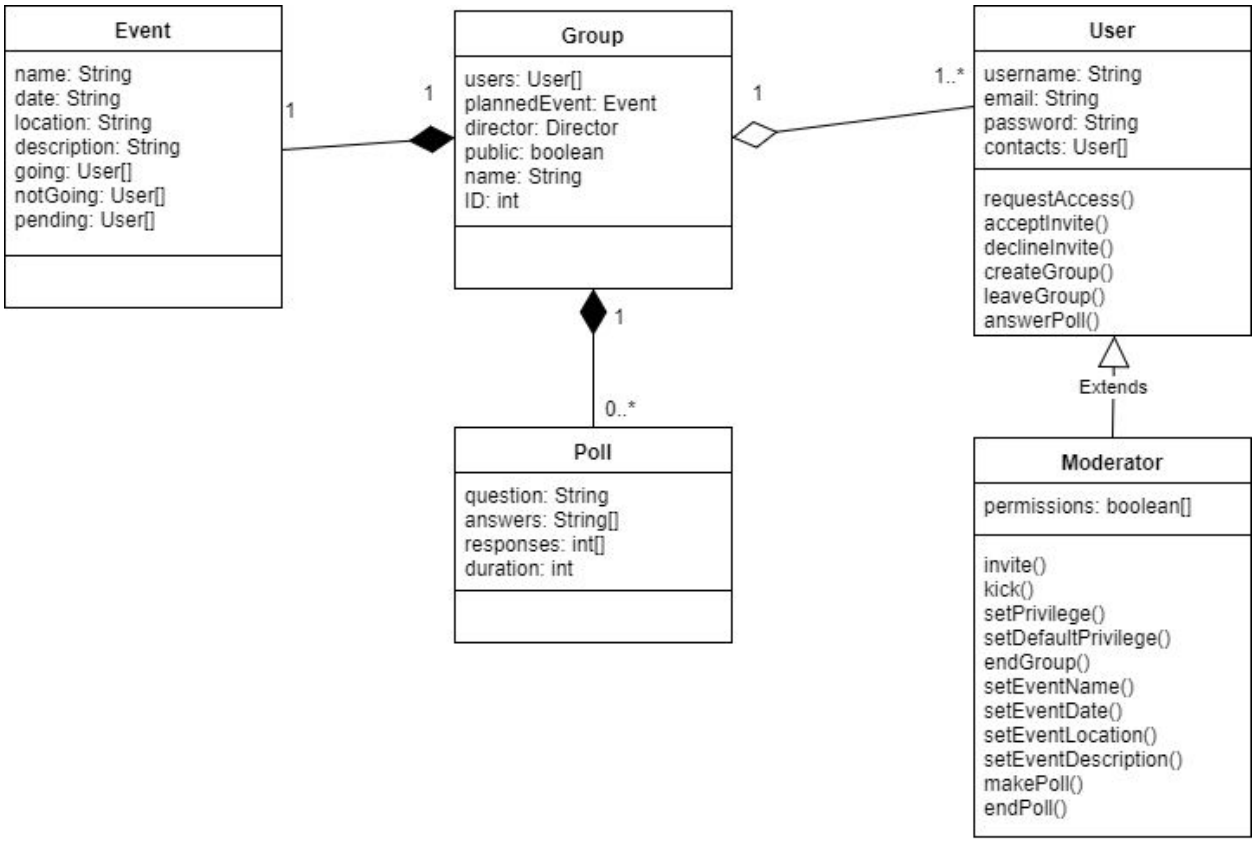
4.1



5. Data View

The data view is concerned with the functionality that the system provides to end-users.

5.1



6. Work Assignment View

The work assignment section details the job responsibilities of each team member working on this project organized by project segment.

6.1

Team Member	Modules	Responsibilities
Christopher Callan	Coding	HTML, CSS, JavaScript Development
	Documentation	Front End Coding Documentation
	Front End Development	Functionality Design and Implementation
Danny Do	Coding	HTML & CSS Development
	Front End Development	User Interface Design/Layout
	Graphic Design	Logo, Images, Etc.
Zach Goad	Coding	HTML, CSS, JavaScript Development
	Documentation	Back End Coding Documentation
	Back End Development	Database Design and Management
Timothy Rine	Team Management	Head Decisions, Team Organizing, Etc.
	Coding	HTML, CSS, JavaScript Development
	Back End Development	Database Management
Oseremen Okajie	Coding	HTML, CSS, JavaScript Development
	Documentation	Front End Coding Documentation
	Front End Development	Functionality Design and Implementation

7. Element Catalog

The element catalog contains a table with all elements used in this SDD and a description of what they mean.

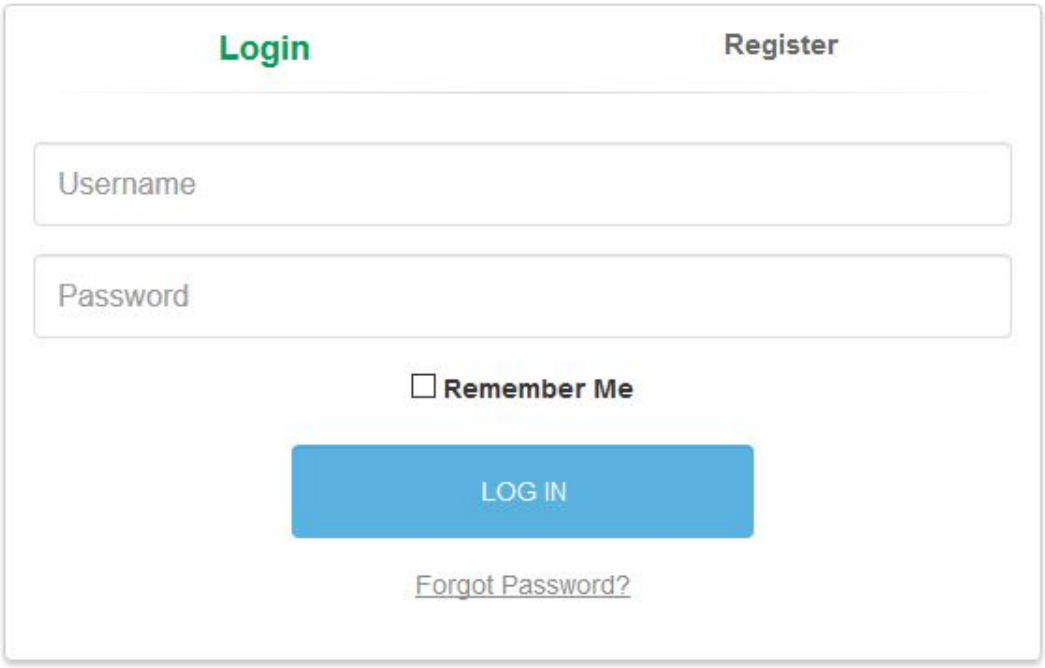
7.1

Hardware Symbol	Description
	This symbol represents a folder. “Label” is the folder’s name.
	System administrator workstation
	User’s computer/phone/tablet
	Public communication network
	Server

8. User Interfaces

In this section you will find sample screenshots of the current user interfaces.

8.1



logo

HOME EVENTS CALENDAR CONTACTS FAQ

Welcome to PlanIt

[Text]

SIGN UP

80 × 85

User Friendly

[Text]

80 × 85

Smooth Design and Interface

[Text]

80 × 85

It's Free!

[Text]

9. Database Design

This section displays an overview of the database design that will be implemented.

9.1