Developer Manual

Chris Network Solutions

11/11/2020

Contents

Developer Guide

1.0 Overview

This Developer Manual for *Chris Network Solutions* documents all the languages and functions utilized in this dynamic website to be able to make updates to the website if necessary. *Chris Network Solutions* is handled through a GitHub repository. The GitHub repository allows the team to maintain version control.

https://github.com/alb-ctrl/CPS-4301-SIT

2.0 Languages

The programming languages used to develop *Chris Network Solutions* is as follows:

- HTML
- CSS
- PHP
- JQuery/AJAX
- MySQL (the database)
- Additional JavaScript Libraries

PHP - planned to implement the guidelines as explained in the link, such as camelCase for functions and meaningful variable names. One of the most important aspects enforced are the utilization of verbs in functions and in If-statements. This is so the functions or variables are self explanatory. https://www.phpdeveloper.org.uk/articles/php-coding-guidelines/

JQuery/AJAX – used together to make our connection to the MySQL database asynchronous, we will follow normal style conventions similar as the ones from the other languages explained. For more information in the link:

https://contribute.jquery.org/style-guide/js/

Developer Guide

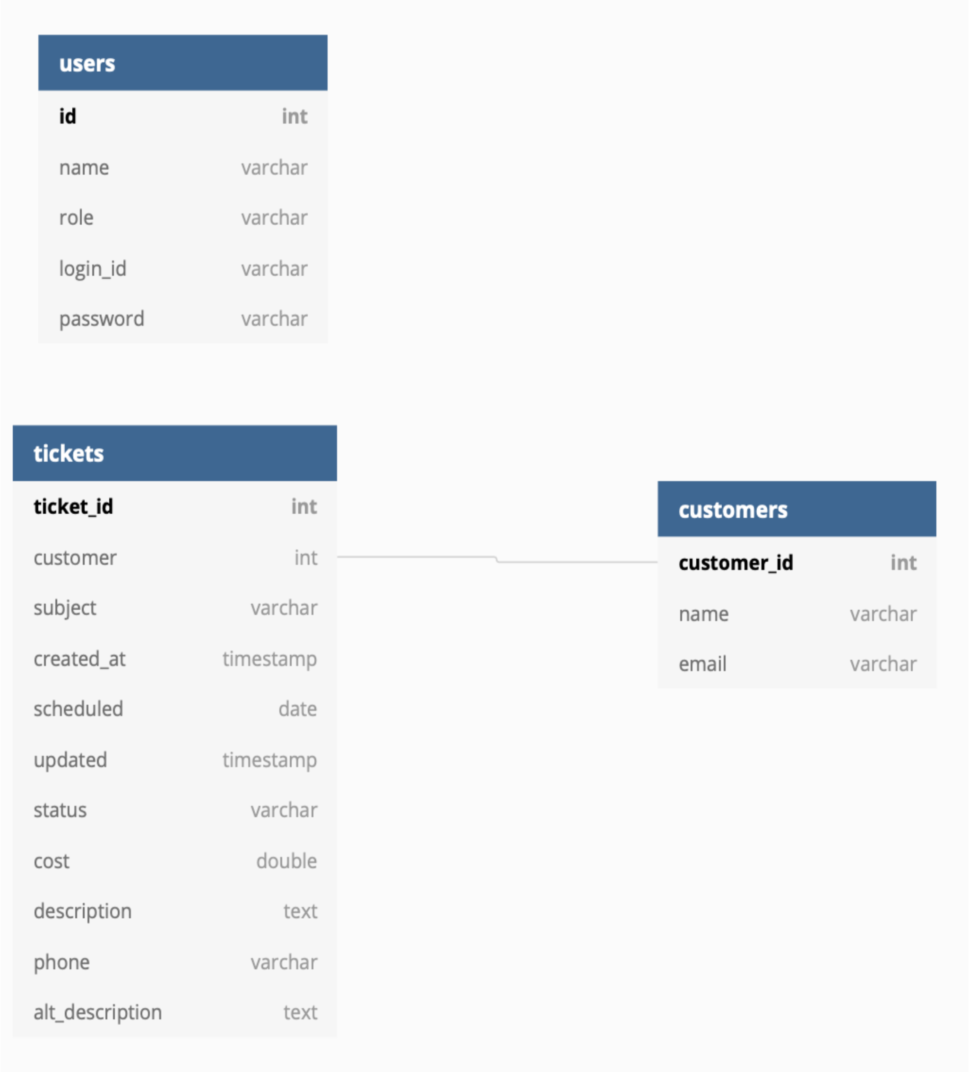
HTML - Extra space is left after section tags and indent. This is to increase readability and avoid typo errors. The language is easily understandable, lightweight, and gives way to tags and labels that are recognizable by all. Simpler development translates into more elaborate pages where we can include content and requirements in a consistent way. http://html-tutorial.eu/html5-style-guidelines-and-coding-standards-html-43

CSS – it is used for its high flexibility and control when considering website specifications, and of course, it's compatibility with HTML. With CSS, the website design is responsive and avoids making it too heavy to ensure faster load times. Two of the most important points we consider and enforce the most when we use CSS is the improvement accessibility and scannability of content, as well as ease of organization and making changes. This caters to the Agile Programming Method and considers the possibility of the client making changes along the development process.

https://cssguidelin.es/

2.1 Database

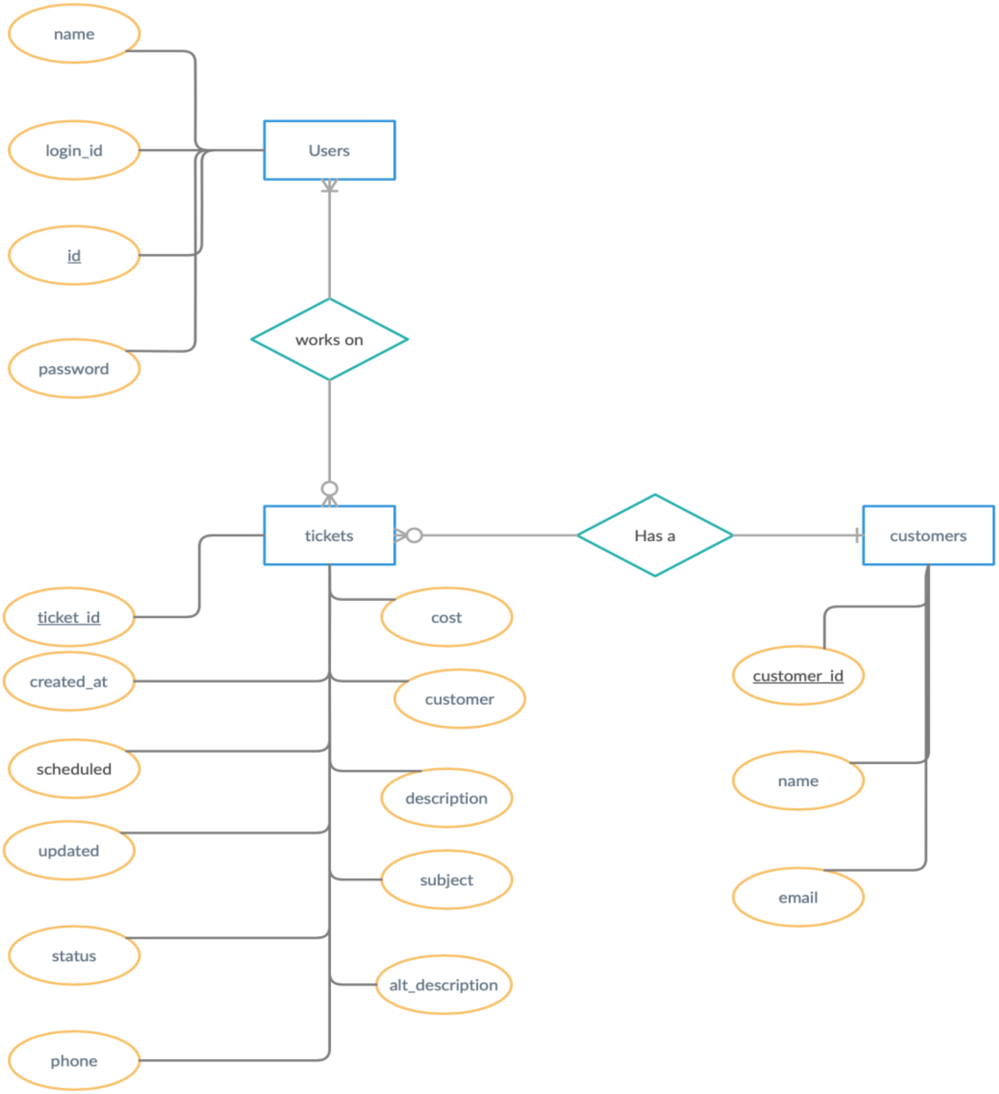
Chris Network Solutions utilizes a MySQL database to store any contact and ticket information. The structure of the database is as follows:



users	
id	int
name	varchar
role	varchar
login_id	varchar
password	varchar

tickets	
ticket_id	int
customer	int
subject	varchar
created_at	timestamp
scheduled	date
updated	timestamp
status	varchar
cost	double
description	text
phone	varchar
alt_description	text

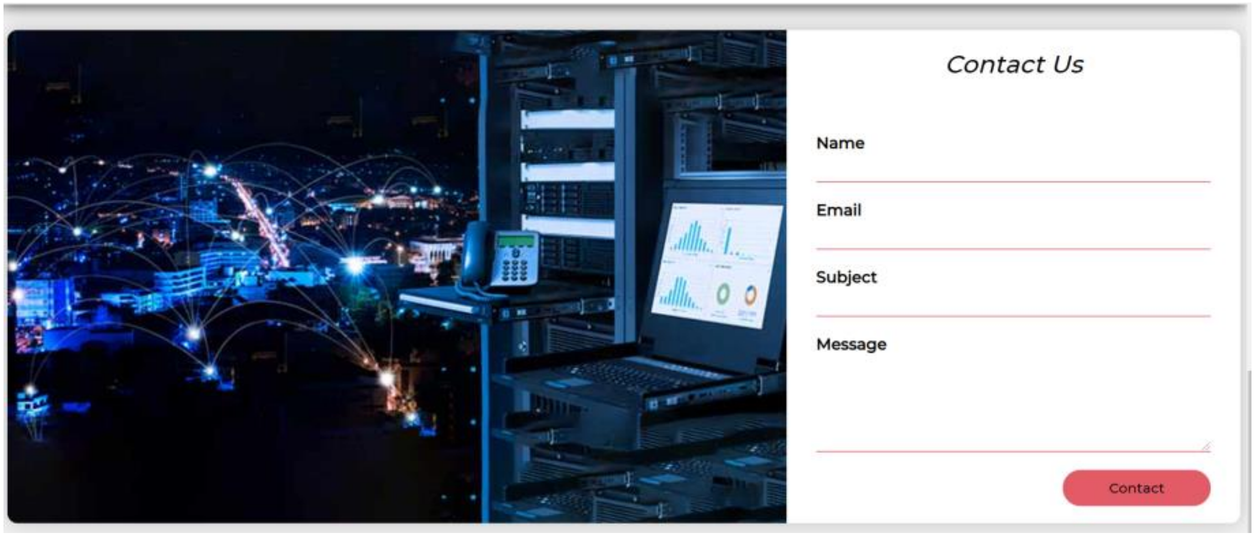
customers	
customer_id	int
name	varchar
email	varchar



For login information, this is stored in the **Users** table, which the account information is generated manually through the database itself.

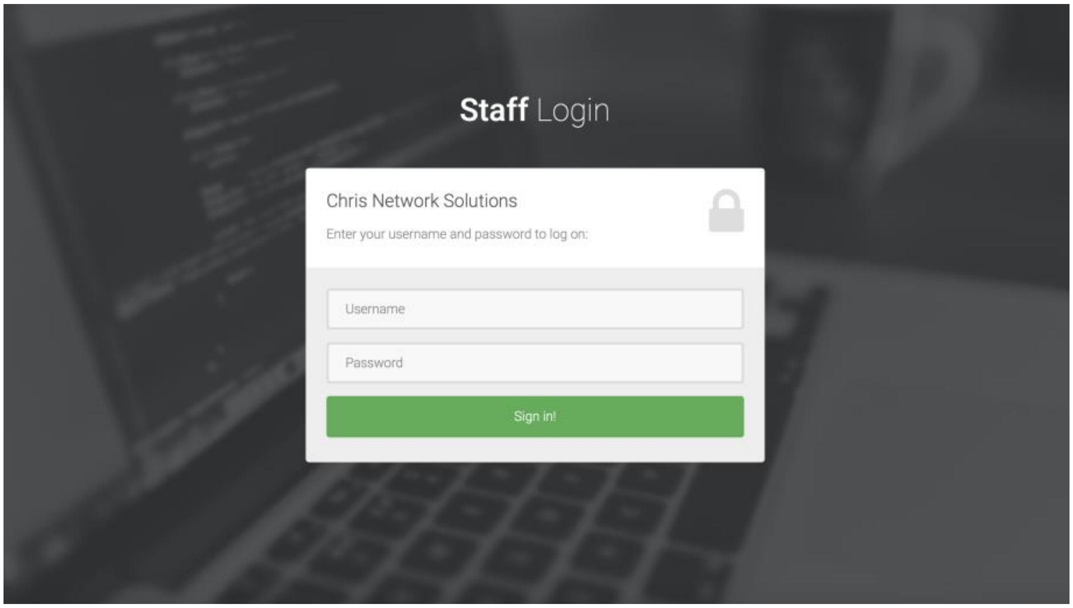
3.0 Screens

3.1 Contact Form



3.2 Login Page

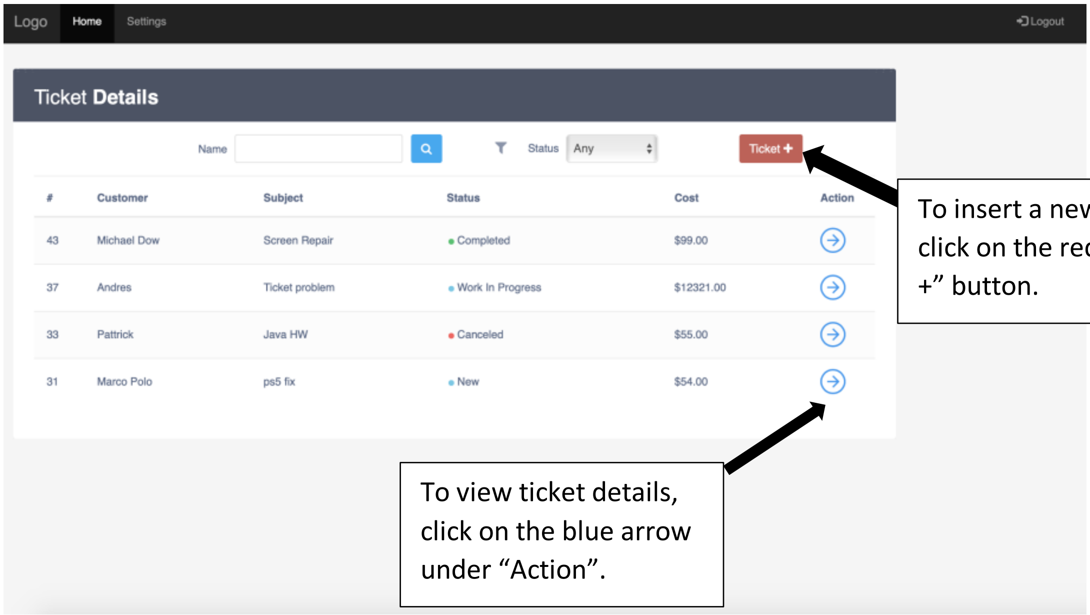
To access the login page, type into your address bar *https://chrisnetworksolutions.com/login*


Staff Login

Chris Network Solutions

Enter your username and password to log on:

3.3 Ticketing System

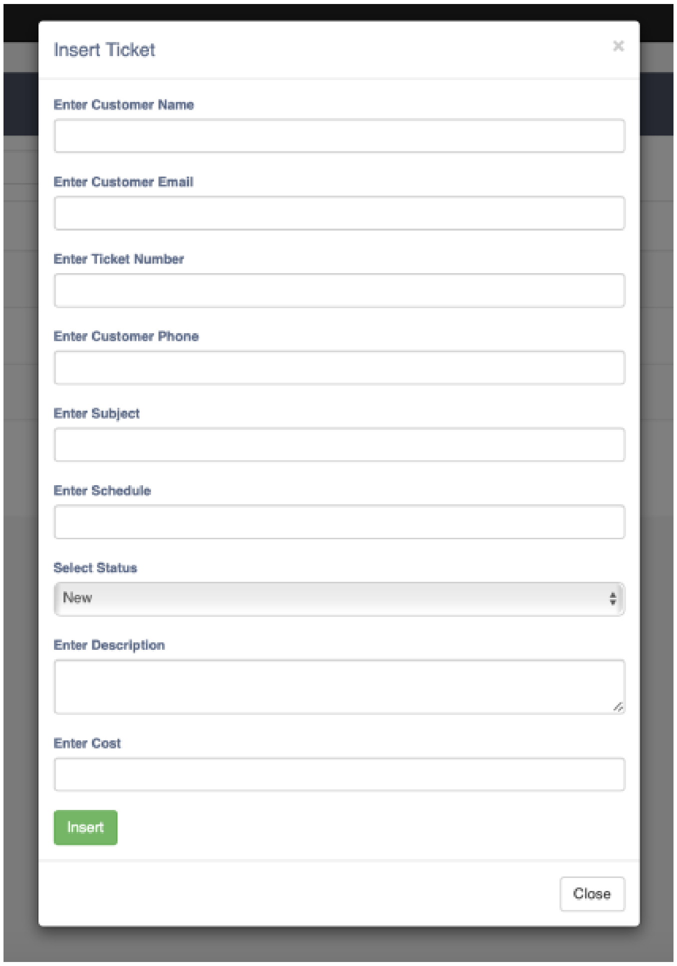


Ticket Details

#	Customer	Subject	Status	Cost	Action
43	Michael Dow	Screen Repair	● Completed	$99.00	→
37	Andres	Ticket problem	● Work In Progress	$12321.00	→
33	Pattrick	Java HW	● Canceled	$55.00	→
31	Marco Polo	ps5 fix	● New	$54.00	→

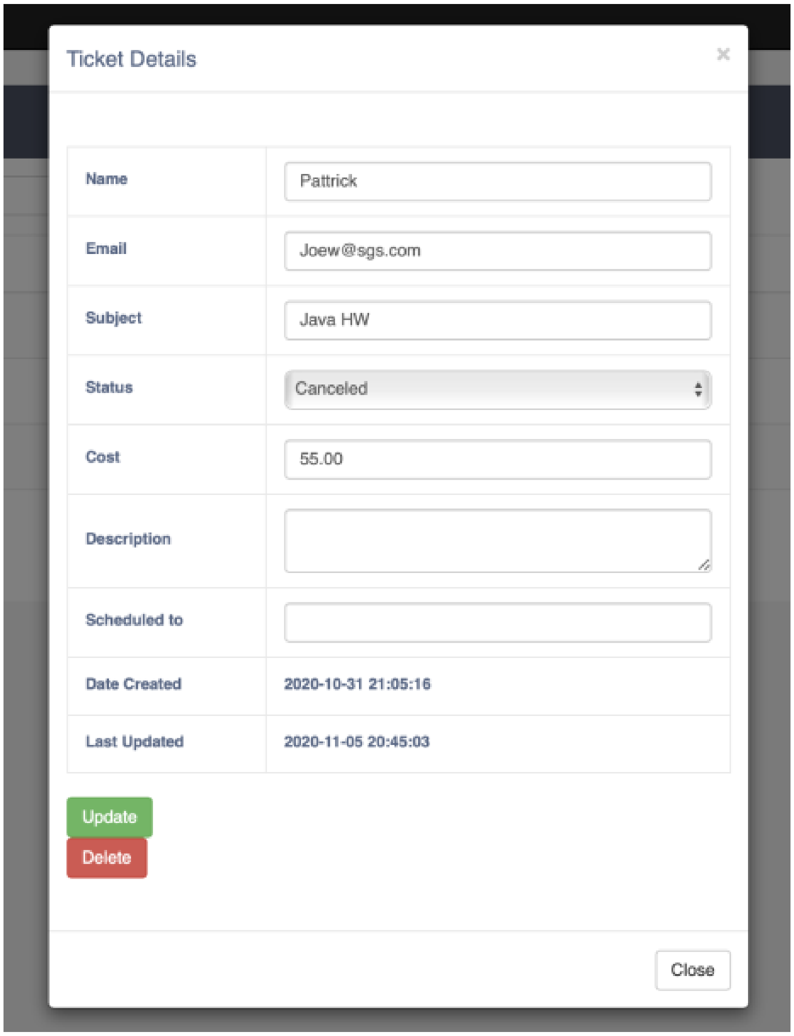
To insert a new ticket, click on the red “Ticket +” button.

To view ticket details, click on the blue arrow under “Action”.



Select Status: New

Developer Guide



Name	Pattrick
Email	Joew@sgs.com
Subject	Java HW
Status	Canceled
Cost	55.00
Description	
Scheduled to	
Date Created	2020-10-31 21:05:16
Last Updated	2020-11-05 20:45:03

4.0 Functionality

Contact Form - The contact form requires the user to fill the textboxes of name, email, and description. There will be verification of the email attribute to make sure the email address is valid. If all textboxes are filled out and validated, the form information will be sent into the database. A customer will be populated in the customers table of our database. In the customers table is the name of the customer and their email. The description will be sent into the ticket table, which would ultimately parallel the ticketing system for our client to fill out the rest of the attributes accordingly.

- Pre-condition - The user will fill out the contact form. This includes their name, email, and description of inquiry.
- Post-condition - If email inputted meets the regular expression conditions, the form will be successfully submitted. The website will output that "the form was submitted successfully."

Login Page - The login page will be a simple login form with a username textbox and password textbox. This login page is to be able to access the knowledge base of Chris Network Solutions, which parallels between the two. The username and password used to access the knowledge is stored in the login table of our database. These credentials will be manually inserted into the database through our database administrator. There is no account needed to be created, since the only people accessing the knowledge base will be our client, Chris, and any administrators.

- Pre-condition - The user, specifically our client and administrators, will input a username and password to access the knowledge base.
- Post-condition - If the username and password inputted matches with the user table database records, the system will redirect the user to the knowledge base UI.

Ticketing System - The ticketing system is where our client will be able to create, update, and delete tickets based on information derived from the contact form that gets sent into the database. The ability to create, update, or delete tickets will be through a button on the UI for our client to click. The customer information is attainable through the knowledge base, which is where the ticketing system is located, too. Our client will be able to fill out the ticket form with the attributes of phone, status, cost, and additional information. Additional attributes such as created_at will be automatically generated with a timestamp of when a customer submits a form to the knowledge base.

- Pre-condition - The client or administrators will create a ticket. This involves inputting information regarding the customer at hand.
- Post-condition - If all of the information inputted in the ticket follows specific conditions, the ticket will be successfully created.