LinkPoint Wiki

Software Design and Development Specifications

Prepared by: Ken Chang p. 604.559.8324 x320 kchang@linkpoint.ca

Table of Contents

I. **Introduction** 2

Overview

Context

Goals and Objectives

II. **Terms and Definitions** 2

III. **General Design Constraints** 3

Production Environment

Mandated Constraints

IV. **General Requirements** 3

Performance Requirements

Security Requirements

Legal Requirements

V. **System Architecture** 5

Overview

Network Architecture

System Diagram

Database Design

Application Component Design

VI. **Functional Requirements** 9

Common Functionality

System Level

Tenant Level

Client Portal

Meetings Module

HR Module

ConnectWise Module

Application Program Interface (API)

VII. **Appendix A – JSON Mapping** 23

Workstations and Logged In Users

Introduction

Overview

The LinkPoint Wiki (“LinkWiki”) is a web-based information management system with the primary function of providing a secure, online, multi-tenant system for use in the MSP industry. LinkWiki will provide storage, management and synchronization of organizational/client credentials and documentation for users of the application.

Context

The purpose of this document is to provide the development team and key stake holders with a guide defining functional requirements, design guidelines, features, and roadmap.

Goals and Objectives

The primary objective of this system is to provide a secure, centralized, online platform where users in the MSP Industry can utilize to manage their organizational and client data.

The application is expected to:

1. Function in a simple and intuitive manner
2. Provide a desktop and mobile friendly web interface
3. Can sync with ConnectWise, Sherweb (and other 3rd party sources), and track changes
    a. i.e.: have plug-in/API capabilities
4. Store data in a secure and encrypted manner
    a. We are open to a block chain based model

The project goals are to have a platform that is

1. scalable; and
2. can be maintained and updated by internal dev teams

Terms and Definitions

Term	Description
LinkWiki	The name of the application itself, also referred to as “The Application”
System Level	Top level admin layer of the application. This level is where the LinkWiki administrators reside to define Tenant admins, contact information, subscription levels and billing.
Tenant	The tenant is the end user of the LinkWiki. Tenants will only have their own configuration, user management, permissions/groups and access to their own data.
Client	A client can be an organization or individual for whom credentials and documentation will be managed for. Clients are associated to individual Tenants.
MSP	Managed Service Provider is a company that manages information technology services for other companies; acting as their technical support department.
RMM	Remote Monitoring and Management system used by MSPs to monitor and manage devices such as workstations, laptops, servers from a centralized console.

AWS	Amazon Web Services
TBD	Denotes a feature or process that has not yet been determined or to be added in a later phase.

General Design Constraints

Production Environment

The LinkWiki application will reside on an AWS EC2 Linux instance that will be publicly accessible with best efforts firewalling and security measures in place. AWS KMS can be used for key management. The initial environment will have the application server and database on the same instance, but the database will eventually move to its own server as application performance requires it.

Mandated Constraints

With the evolution of the application, additional decisions may be made while in the process of development.

1. The application is to be written in Python on a framework that supports growth and scalability
2. Modals are to be avoided except for information display purposes (i.e. deletion confirmations) or where necessary.

General Requirements

Performance Requirements

The application should operate with minimal delay and any need for loading additional pages wherever possible. There are no current estimates on the number of concurrent users but the application should support a minimum of 200 users at one time. Scalability and load balancing architecture is to be defined during the development process.

Security Requirements

Only authorized users with the appropriate permissions will have the ability to access anything on this application. It is assumed that all actions outlined in this document (unless otherwise noted), are performed after successfully completing the authentication process.

Due to the nature and sensitivity of the data stored in the application, data at rest must be encrypted via AES-256 encryption with a key unique to each tenant and stored in a secure manner. The key is only to be accessed when data encryption/decryption is requested by the application. Keys are not to be stored in the database.

User login passwords are to be salted and hashed.

Login attempts will trigger a temporary lockout (10 minutes) with an override available to administrators via user management.

Multi factor authentication will be required for all users (System and Tenant Level). At present only Authy (https://authy.com/) will be supported as a 2FA provider. Additional providers will be added later but the capability and framework is to be built now.

Only one (1) session per user is to be allowed at a time. If a new session is created on another device, the pre-existing session is to be destroyed and logged out immediately.

Legal Requirements

(*Refer to Security Requirements*)

System Architecture

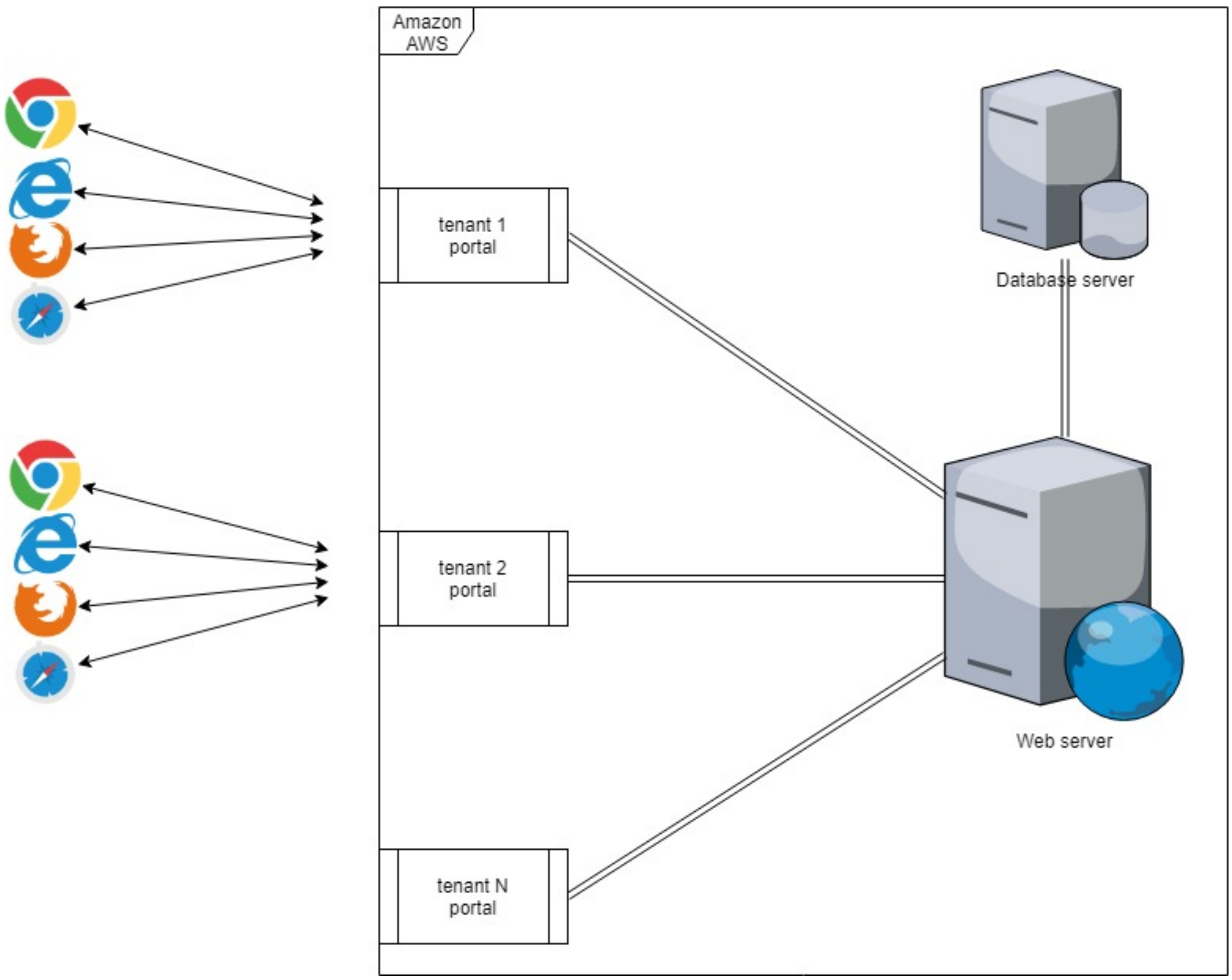
Overview

The LinkWiki will poll ConnectWise via API (unique API keys and hosts per tenant) at a regular interval for the purpose of synchronizing stored usernames and passwords and matching users to workstations and devices. (*See ConnectWise Module for further details*)

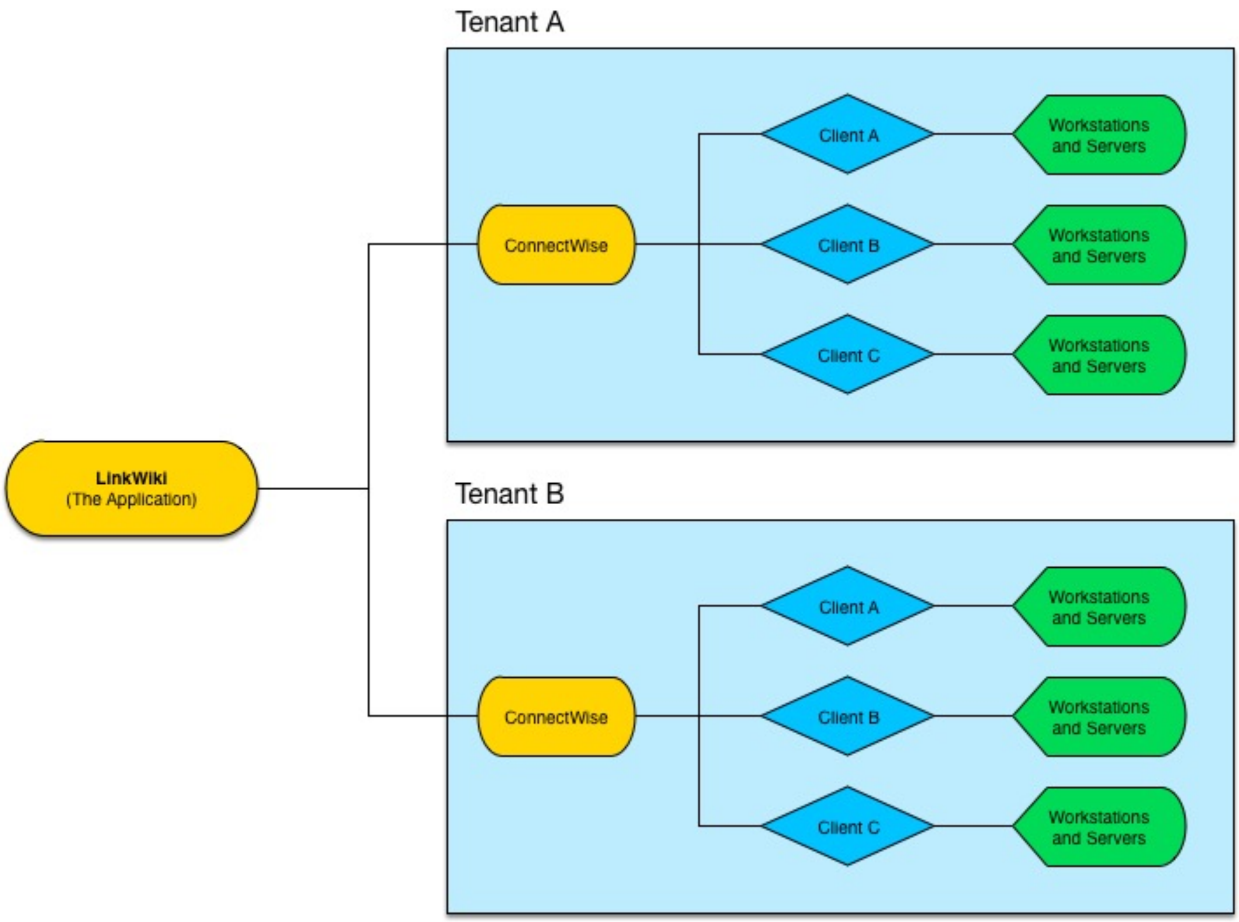
Network Architecture

Users accessing the application will be via various web browsers and can be through various entry points.

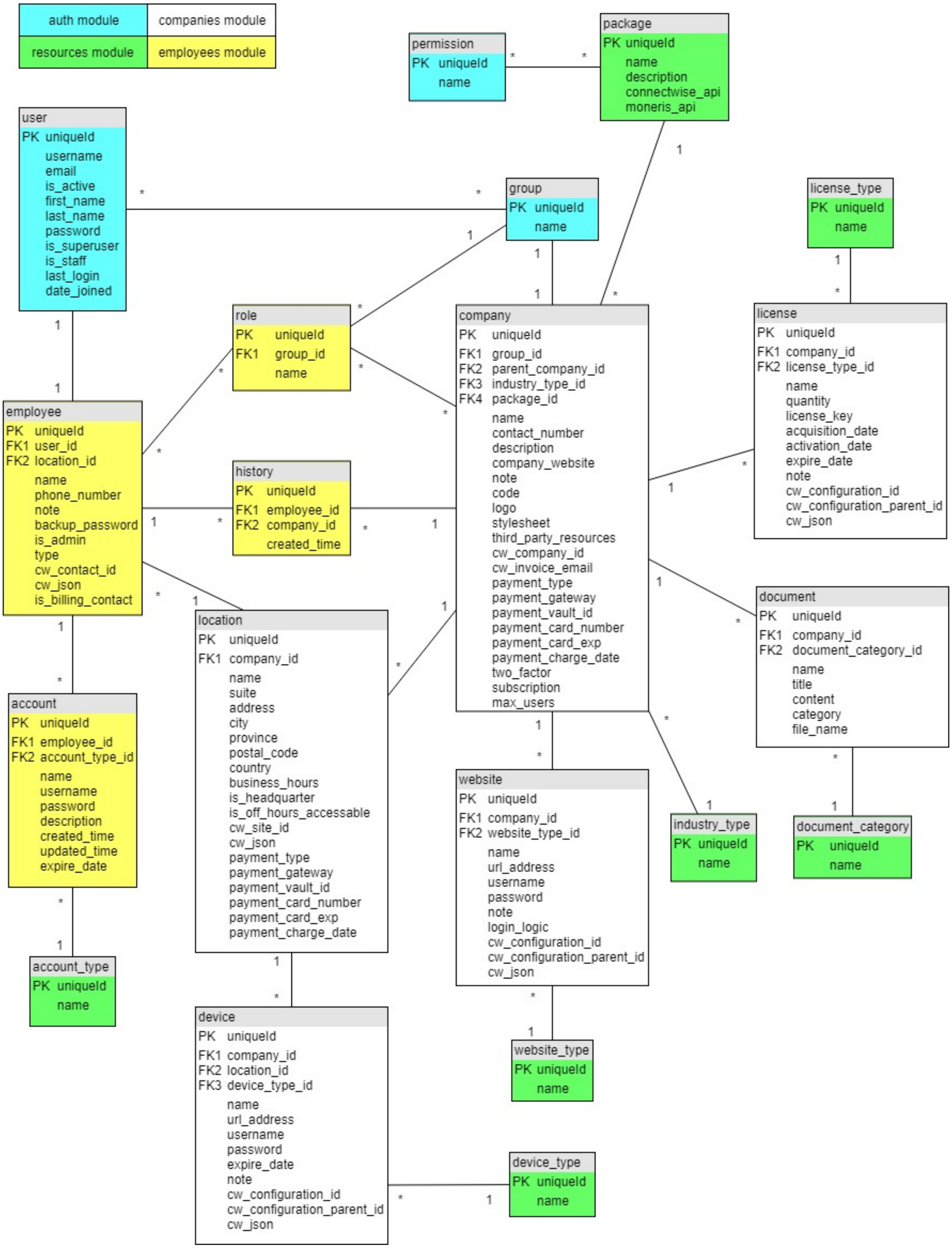
- <tenant id>.applicationurl.com
    - e.g. bigcompanyco.applicationurl.com
- applicationurl.com
    - login form will have a field for tenant id, username and password



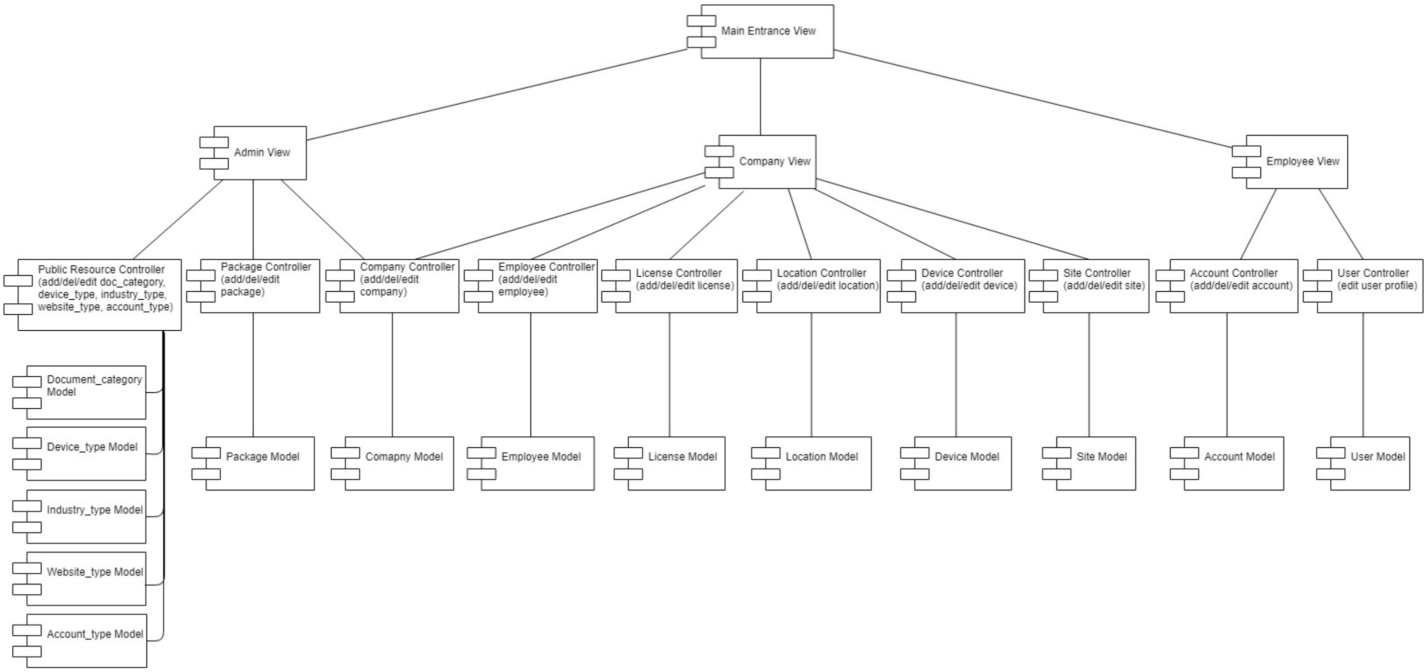
System Diagram



Database Design



Application Component Design



Functional Requirements

The following describes the areas of the LinkWiki and expected functionality.

Common Functionality

Editing

All editing will be performed in-line unless otherwise specified or unfeasible.

Deleting

Deletion of data should always prompt the user for confirmation by typing “YES”

Logging

All actions by users will be logged with their user id, description of action, date/time action took place and any other relevant information (i.e. what was changed)

System Level

Security Levels

Ability to create users and roles for system level access. Roles can be defined and customized based on the following access permissions.

Permission	Access Description
Admin	Full access (all permissions checked)
Create Tenant	Ability to add a tenant
Modify Tenant	Edit an existing tenant
Delete Tenant	Delete an existing tenant
View Billing	View current billing/balance/invoicing
Delete Billing	Delete/write off an invoice
Create Billing	Same as ability to create tenant
Create Package	
Modify Package	
Delete Package	

Tenant Management

Create Tenant

Type: Use case
Level: System Level
Permissions: Create Tenant, Admin

Description

This process is for when a system level user with the permission to create a tenant submits a registration form.

Process

- Page displayed to provide contact information for tenant user (fields below)
- If there is missing information, an error message is displayed indicating what still needs to be provided
- Username (email) and password is generated by the system
- Welcome email is sent (if selected) to Admin contact with link to reset password
- Also logged is the user who created the account, and date/time stamp
- User is returned to the Tenant list and acknowledgement is displayed

Fields

Contact Information

- Company Name (string:255)
- Address 1, Address 2 (string:255)
- City (String:40)
- Province/State (String:80)
- Postal/Zip Code (String:10)
- Country (String:40)
- Admin Contact First Name (string:50)
- Admin Contact Last Name (string:50)
- Admin Contact Email (string:60)
- Admin Phone Number (string:15)
- Admin Phone Number Extension (int:8 - not required)
- Send Welcome Email (checkbox)

Subscription

- Assign Subscription (dropdown selection)
- Billing Contact First Name (string:50)
- Billing Contact Last Name (string:50)
- Billing Contact Email (string:60)
- Number of users (integer)
- Discount/Price Override
    - Per user

Add-Ons

- HR Module
- Meetings Module
- *More TBD*

CRM

- Notes (text field)

View/Modify Tenant

Level: System Level
Permissions: Create Tenant, Modify Tenant, Admin
Mockup: http://

Description

This process outlines the activities involved and option when a system level user clicks on an existing tenant to view or modify their information.

Process

- Initial view will only display information about the tenant
  - Contact Info
  - Subscription (plan, pricing, and usage: x users out of y licensed)
  - Current balance
  - Notes field
- Button to edit / delete tenant (if permitted)
- Button to go to billing area (if permitted)

Delete Tenant

Level: System Level
Permissions: Delete Tenant, Admin
Mockup: none

Description

This process is when a system level user with the permission to delete tenants clicks the delete button for a specific tenant.

Process

- When delete button is clicked, prompt for confirmation (by typing "YES")
- Check for open billing
  - If Administrator-level user, confirm deletion of account. (by typing "YES" again)
  - If not, stop and notify user that deletion requires Admin.
- Present option for exporting data to PDF prior to deletion
  - If yes, prompt for contact to email PDF to (current system level admin or select from list of administrator users on tenant account. Then generate PDF and email.
- Delete all account data

Tenant Search

Level: System Level
Permissions: Any
Mockup: none

Description

This process is when a system level user begins typing in the live search box on the tenant list for either a tenant name, or phone number.

Process

- While typing in the search field, begin reducing the matching results in the tenant list table
- If not matches found, return “no matches found”

Package Management

Create Package

Level: System Level
Permissions: Create Package, Admin
Mockup: http://

Description

This process is when a system level user with the permission to create packages submits the creation form.

Process

- When the form is submitted, ensure required fields (Package Details) have been entered
- If not, notify user of missing fields
- Check if Package Name already exists and notify user if it does
- Save Package to database with ID, created by, and date/time stamp
- Return to Package list

Fields

Package Details

- Package Name (string:100)
- Description (text)

Options and Features

- Moneris Module (checkbox)
- ConnectWise Sync (checkbox)

View/Modify Package

Type: Use case
Level: System Level
Permissions: Create Package, Modify Package, Delete Package, Admin

Description

This process outlines the activities involved when a system level user clicks on an existing package to view or modify subscription information.

- Initial view will only display information about the package
  - Package Name
  - Description
  - Count of tenants using this package
  - Feature options
- Button to edit / delete package (if permitted)

Process

- If the Edit button is clicked, Package details should become inline editable

Delete Package

Type: Use case
Level: System Level
Permissions: Create Tenant, Modify Tenant, Admin

Description

This process outlines the activities involved and option when a system level user clicks on an existing tenant to view or modify their information.

Process

- When delete button is clicked, check to ensure the package is not being used by a tenant account
    - If not, prompt for confirmation (by typing "YES")
    - If it is, display message notifying user that package is being used and cannot be deleted. Stop
- Delete package

Tenant Level

Security Levels

Permission	Access Description
Admin	Access to manage users and application settings
Accounting	Ability to access accounting module
HR	Access to HR module
User	General User

Application Settings

Integrations

This page will list all integrations installed by the tenant. The user can click on an integration name to edit any settings related to the integration. Integrations should be modular and simple to add. Eventually a market place may be built to extend this.

ConnectWise integration, will require the following fields. There will also be a toggle to pause syncing the two systems as well as a button to “Sync Now”

- FQDN of ConnectWise Host
- Company Name
- Integrator Login ID
- Integrator Password

Moneris Credit Card Processing integration:

- API key

Appearance

Description

Basic settings for the UI of the application per tenant.

- Company Logo
- Basic colour scheme

Categories

Description

This area manages the categories used to identify account types when recording credentials and documents.

- Account Categories
- Document Categories (*see Company: Document List*)
- Device Categories (*see Company: Devices*)
- Industry Categories

User Management

Description

Management of tenant's users. Upon submitting new user information, an invitation email is sent to the user to set a password.

- First Name
- Last Name
- Email Address (Username)
- Department
- Client Group

If the HR module is installed and enabled, additional fields will become available as a new section

- Start Date
- Title
- Contact Number
- Birthdate
- Notes
- Emergency Contact Information
    - Name
    - Relationship
    - Contact Number

Client Group Management

Description

Define user access permissions to specific client organizations.

- List of client organizations with a checkbox
    - Checked denotes user has access
    - Unchecked denotes user has no access
        - Save each change on click

Templates

Description

Allows an administrator to create templates for the tenant organization to be used within the application based on the template type.

Template types can be:

- Document
- Meeting
- Incident Report
- HR (If the HR module is enabled)

Templates will utilize a custom keywords system as variables to be replaced upon loading the template in the application.

Keyword	Description / Action
%%date%%	Replaced by current date (formatted: mmm dd, yyyy)
%%time%%	Replaced by current time (formatted: hh:mm [a/p])

User Profile

Personal Settings

- Profile image
- Change password
- Number of results per page (dropdown of 25, 50, 100, 200, All)

Client Views (Companies)

Live Search

Description

Live name search capability to filter client table list as user types. This search field should take focus and select all text on load at all times. Returning to the list should have the previous search term still in the box and filtered.

Adding a Company

Description

Adding a company opens a new page with a form for the following contact/site information

- Company Name
- Site Name
- Address 1, Address 2
- City
- Province/State
- Postal/Zip Code
- Country
- Main Phone Number
- Fax Number
- Website
- Logo (attachment)
- Active Directory Domain (string:optional)

Optionally, an additional site address can be provided by clicking “Add Site” which will prompt for the same information. By default, the information entered will create a site entry with the *Site Name*.

Company: Home View

Mockup: https://app.moqups.com/VjPWIbPHHh/view/page/ad64222d5

Description

This is the initial view upon clicking on a Company name from the list or search results. Side navigation shows the preset categories of data stored in the LinkWiki and the number of entries in each (there are 5 documents for Big Company Co.).

Client site information is shown on the initial view with their logo (if not provided, a placeholder with the initials of the company will be shown, i.e. "BCC"), location address and phone number, website and a small Google Maps preview.

Recent activity will list changes and entries that have been recently made to the client records. Only users with Admin level access will show who made the changes in the activity list (indicated by "- by <user's name>")

Company: Document List

Mockup: https://app.moqups.com/VjPWIbPHHh/view/page/a7b79c083

Description

List of documents entered for a client with live search capability. A button to create a new document with document template drop down located at the upper right.

Company: Document View

Mockup: https://app.moqups.com/VjPWIbPHHh/view/page/a64e49ff2

Description

Upon opening the document, the content should display with the title, last updated date and last author. The right navigation will also display an edit button, a button to share a link to this document (not shown on mockup), file attachments, linked documents and information about revision history.

Company: Adding/Editing/Deleting A Document

Description

When creating a new document, load the template selected in the prior step. Interface will show the following fields. Uploaded files and links will have a delete button next to each entry to remove them from the document. All changes (editing, adding/removing attachments, adding/removing linked documents) are to be logged with an option to rollback changes by admins only.

- Title
- Document Type selector
- Simple WYSIWYG editor for content entry
- List of Client sites and checkboxes for selection
- Field to upload multiple files

- Field to live search all documents belonging to the tenant to allow linking to related documents in other companies.
    - Allow linking to multiple documents
    - Search results should be in the format: *Company Name – Document Title*

Company: Employees List

Mockup: https://app.moqups.com/VjPWIbPHHh/view/page/a2556863f

Description

Default view of the employee list will only display the name, phone number, email address, location and workstations list.

Default sorting will be grouped by Active Status, followed by Employee Type (VIP, Employee, 3rd Party), then sorted by Employee Name.

Workstation Matching: Matching is performed via ConnectWise AP module. (*see ConnectWise Module*)

Clicking on a name will expand the details for the employee and display entered credentials in card format. Passwords are always hidden unless clicked on to display the text. Clicking the copy icon next to a username or password will copy that value to the clipboard.

Edit icons available for all records for inline editing and delete icons with confirmation to delete.

Company: Add Employee

Description

Clicking the Add New Employee button will display a form at the top of the page to enter Employee information. Submitting the form will add the employee to the list without reloading.

- Employee Name
- Email Address
- Phone Number
- Location (site dropdown)
- Employee Status
    - Active
    - Inactive
- Employee Type
    - Employee
    - 3rd Party
    - VIP
- Short Note
- Portal Access

Company: Devices List

Description

For the purpose of storing device credentials.

Default sorting is to be Location, then Device Type

Company: Adding Devices

Description

Devices will have the following fields

- Device Type
- Description
- IP Address/URL
- Username
- Password
- Location (site dropdown)
- Notes
- Asset Tag Number (ConnectWise Module only)

If ConnectWise is enabled/installed two-way synchronized between the LinkWiki and ConnectWise hardware configurations will be performed.

Company: Software Licenses

Description

Software licenses will have the following fields

- Software Name
- Serial Number
- Install Date
- Location (site dropdown)
- Software Asset Tag
- Workstation Asset Tag

Other Services

Description

Credentials stored here are for 3rd party services/applications. Fields for this area are:

- Service Type
- Description
- Username
- Password
- Website

- Notes

Client Portal

A portal available to clients will allow employees (with portal access enabled) to view/add/manage their own credentials only.

Meetings Module

This module will facilitate documenting meeting minutes and tasks.

Functionality

New Meeting

Description

Upon creating a new meeting, the user will be prompted to provide a title, and enter the names of the attendees. Typing names in the attendees field will bring up a results list of users belonging to the tenant which can be clicked to auto complete.

WYSIWYG editor also provides functionality to @ tag users regardless of their attendance.

View Meeting Minutes

Description

Viewing meeting minutes allows the user to download a PDF copy of the minutes or email to the user or all attendees.

HR Module

This module allows management to record employee activities (incident reports, reviews, etc)

Functionality

Employee List

List all employees with their start dates

Employee Detail

The Employee Detail page will display Employee information and documents attached to the user

New documents can be added utilizing the HR templates. (This interface will be similar to the Meetings Module)

ConnectWise Module

While currently referred to as the ConnectWise Module, this portion of the application will provide integration support for other RMM's and 3rd party sources .

Configuration Wizard

The first time this module is configured by an administrator, the LinkWiki will walk the user through the setup process. This includes prompting for the API login information (and testing the connection), performing and initial sync (with mapping confirmation), and setting additional options.

Data Synchronization

Companies

This page will allow an administrator user to map companies from ConnectWise to those in the LinkWiki if they pre-exist. Fields should be:

- LinkWiki Company Name
- ConnectWise Company Name (Dropdown with best match selected)
- Name Action (dropdown)
    - Use LinkWiki Name – Rename the Company in ConnectWise to the name in LinkWiki
    - Use ConnectWise Name – Rename the Company in LinkWiki to the name in ConnectWise
    - Map Only – No changes to either side, simply map

Scenarios

Ignore/Do Not Sync – Ignore this company

Application Program Interface (API)

Outbound Interface

The LinkWiki will provide an API interface for internal use initially for mobile applications and potential other development.

The API interface will provide functions for accessing/adding/modifying tenant companies and all stored information for that company.

Inbound Interface

The inbound interface will provide an adapter for accessing 3rd party APIs at regular intervals along with as needed interactions with 3rd party applications.

Appendix A – JSON Mapping

Workstations and Logged In Users

JSON


```
{
    "id": 7004,
    "name": "A1-21537",
    "type": {
        "id": 24,
        "name": "Managed Workstation",
        "_info": {
            "type_href": "https://api-
na.myconnectwise.net/v4_6_release/apis/3.0/company/configurations/types/24"
        }
    },
    "status": {
        "id": 2,
        "name": "Active",
        "_info": {
            "status_href": "https://api-
na.myconnectwise.net/v4_6_release/apis/3.0/company/configurations/statuses/2"
        }
    },
    "company": {
        "id": 20105,
        "identifier": "A1 Pallet Enterprises",
        "name": "A-1 Pallet Ent.",
        "_info": {
            "company_href": "https://api-
na.myconnectwise.net/v4_6_release/apis/3.0/company/companies/20105"
        }
    },
    "contact": {
        "id": 4188,
        "name": "Sarah Barrell",
        "_info": {
            "contact_href": "https://api-na.myconnectwise.net/v4_6_release/apis/3.0/company/contacts/4188"
        }
    },
    "site": {
        "id": 1904,
        "name": " Aldergrove",
        "_info": {
            "site_href": "https://api-
na.myconnectwise.net/v4_6_release/apis/3.0/company/companies/20105/sites/1904"
        }
    },
    "locationId": 33,
    "businessUnitId": 1,
    "deviceIdentifier": "2004",
    "serialNumber": "2UA1421M6K",
    "modelNumber": "HP Compaq 6200 Pro MT PC",
    "tagNumber": "21537",
    "installationDate": "2016-09-22T03:21:35Z",
    "vendorNotes": "",
    "notes": "",
    "macAddress": "2C-41-38-8F-36-1F",
    "lastLoginName": "PALLETLAND\\sarah",
    "billFlag": false,
    "backupServerName": "",
    "backupProtectedDeviceList": "",
    "ipAddress": "192.168.1.211",
    "defaultGateway": "184.69.95.42",
    "osType": "Microsoft Windows 7 Professional x64",
    "osInfo": "6.1.7601 Service Pack 1",
    "cpuSpeed": "3100",
    "ram": "3984",
```

```
    "localHardDrives": "     Free           Total  \r\nC: 389274 MB   469580 MB  \r\nD: 876 MB   7255 MB
\r\nE: 0 MB   0 MB  \r\nF: 9873 MB   30935 MB  \r\n",
    "activeFlag": true,
    "managementLink": "labtech:open?computerid=2004",
    "remoteLink": "",
    "mobileGuid": "689a46cd-8afb-46eb-84f3-913799a29567",
    "_info": {
        "lastUpdated": "2019-04-05T07:13:51Z",
        "updatedBy": "CW Automate",
        "dateEntered": "2016-09-15T08:27:06Z",
        "enteredBy": "LabTech"
    },
    "companyLocationId": 33
}
```

JSON Keys of Interest

No.	JSON Key	Description
1	"name": "A1-21537"	Workstation name (display only)
2	"id": 20105	Company ID in ConnectWise (potential mapping point)
3	"lastLoginName": "PALLETLAND\\sarah"	Username of last user logged in to workstation

Use Cases

1. The workstation name is to be displayed within user cards to show which workstations a particular user is logged in to.
2. To be used to match with a client in the LinkWiki. A mapping table will be required to associate LinkWiki Company IDs against ConnectWise Company IDs
3. To be used to match against a specific Employee within a client list.
    a. *PALLETLAND* is the client's Active Directory domain
    b. *sarah* is the username