# Home

Information that can be shown at a glance can be put on the home page

## Bulletin Board

The board will have a standard function that will be displayed if the user isn’t logged in and a user specific option that will display important information related to the user.

### User-Specific

This will display information related to the user and include a link to the related page, then show a random picture or show upcoming events when possible. A max of 5 items should be displayed on the board, however, all items with a priority of 1 or below will be displayed. Items will be displayed and found in order of priority.

The Carousel should include images that are of the same size

|  |  |
| --- | --- |
| Name | Priority |
| Complete Registration | 0 |
| Initial Deposit | 0 |
| Maintenance Fee | 0 |
| Show info on resident orientation | 1 |
| Show Active Community Issues | 4 |
| See Clubs the user is involved with | 5 |
| Show upcoming events the user registered for | 5 |
| Show events the user might be interested in | 10 |
| Show clubs the user might be interested in | 11 |

### Standard

Show random about me information relating to the facility

## Login Incentive

This section is only shown when the user isn’t logged in. There are two sections of equal length that will show either logging in or registering. Both will be show as card groups using bootstrap

## Voting

This is only available to user’s that are residents. This JavaScript enabled section will show all available polls. Polls are divided by community issues and club votes. Voting done on the home screen cannot display votes from past polls or will show a distinction between club or community polling.

## Database Structures

Three new tables should be created for the Home Screen to function along with two other new tables to be used to find polls on the account page.

### Polls

This table should only hold the name and choices of the poll

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | Allowed Null | | | |
| Poll ID [Unique] | Poll Name | Choice 1 | Choice 2 | Choice 3 | Choice 4 | Choice 5 | Choice 6 |

### Poll Responses

This table holds the response to a poll and allows for a custom response, a standard response, or no response.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Candidate Key | | Allowed NULL | Allowed NULL |  |
| Poll ID | Resident ID | Response | Custom Response | Final Response Time |

### Poll Ship

This table controls which residents gets a poll. Polls can be distributed to members of a club or sent to a specific resident. Event ID can be used to ship a poll to residents that are participating in an event.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Allowed Null | Allowed Null | Allowed Null |  |  |
| Poll ID | Resident ID | Club ID | Event ID | Start Time | End Time |

### Clubs

Clubs are a collection of residents that have a structure of leadership and can hold meetings. Clubs may have up to 10 roles, virtually unlimited membership, and set a meeting time and place for each day of the week.

|  |  |  |
| --- | --- | --- |
| Unique |  |  |
| Club ID | Club Name | Creation Date |

### Club Membership

|  |  |  |
| --- | --- | --- |
| Candidate Key | |  |
| Club ID | Resident ID | Role ID |

### Club Roles

This defines the name of the role and the permissions the role gets. To calculate the total permissions a user has look at the points per level: level 1 = 1, level 2 = 10, level 3 = 100, etc. Ranks are ordered because of certain permissions.

|  |  |
| --- | --- |
| Permissions | |
| Level | Authority Granted |
| 1 | Ability to Invite new users |
| 2 | Ability to kick users |
| 3 | Ability to change some club info |
| 4 | Ability to set meeting times |
| 5 | Ability to change a user to a higher rank (up to the user’s role) |
| 6 | Ability to demote a user’s rank (must be below the user) |
| 7 | Ability to Set & Control Polls |
| 8 | Ability to Set & Control Events |
| 9 | Ability to **create roles** at a rank lower than the user |
| 10 | Ability to **change rank permissions** at **every level** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Unique | Candidate Key | |  |  |
| Role ID | Club ID | Role Rank | Role Name | Role Permissions |

### Club Meeting

If a meeting is set as override, then if a meeting is or is not set for that day the overridden meeting day will take president.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Unique |  |  |  |  |  |
| Meeting ID | Club ID | Meeting Place | Meeting Day | Start Time | End Time |

### Events

Events may be associated with or without a club. Events can be managed by induvial residents

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Unique | Allowed Null |  |  |  |  |  |
| Event ID | Club ID | Event Name | Meeting Place | Meeting Day | Start Time | End Time |

### Event Assignments

Assignments are meant to place residents into roles for events. Assignments are completely different from roles. Assignments should have the ability to be randomly assigned; random assignment should first select the specific assignment, roles to randomly pull from, and the percentage of the total roles should be assigned to the assignment. A second type of random assignment can occur with selecting multiple assignments, selecting all roles to be available for the assignments, then evenly distributing between each assignment either with or without regard to roles.

* Used to Separate Teams before events
* Used to Assign sub events that run concurrently
* Used to design queues or checking in

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| Assignment ID | Event ID | Resident ID | Queue ID | Complete? |

Type should be used to know if an ordered queue is necessary

* 0 = Necessary

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Assignment ID | Event ID | Assignment Name | Assignment Type | Assignment Description |  |  |

### Event Roles

|  |  |
| --- | --- |
| Permissions | |
| Level | Authority Granted |
| 1 | Ability to Invite new users |
| 2 | Ability to kick users |
| 3 | Ability to change some event info |
| 4 | Ability to set new assignments |
| 5 | Ability to change a user to a higher rank (up to the user’s role) |
| 6 | Ability to demote a user’s rank (must be below the user) |
| 7 | Ability to Set & Control Polls |
| 8 | Ability to make Procedures |
| 9 | Ability to **create roles** at a rank lower than the user |
| 10 | Ability to **change rank permissions** at **every level** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Unique | Candidate Key | |  |  |
| Role ID | Event ID | Role Rank | Role Name | Role Permissions |

### Event Membership

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Candidate Key | | |  |  |  |
|  | Allowed Null | Allowed Null |  |  | Allowed Null |
| Event ID | Resident ID | Club ID | Role ID | Checked In? | Score |

### Events Stored Procedure

This table is used to populate certain assignments after all participants have been checked in or used to create an ordered queue before participants arrive. When an assignment is about to begin, any member with the appropriate permissions may use a stored procedure then manually adjust the outcome. This will only take in members that are already checked in.

* Random Types
  + 1 = Distribute randomly with respect to roles
  + 2 = Distribute randomly
  + 3 = Distribute while same assignment ID (if a resident has multiple of the selected assignment IDs, then only the first found will be used)
* Distribution Percentage
  + Specify from the available population, the percentage or how many will be selected
* Distribution Order
  + 1 = Random
  + 2 = Alphabetically [Ascending]
  + 3 = Alphabetically [Descending]
  + 4 = Score [Ascending]
  + 5 = Score [Descending]
  + 6 = Rank [Ascending]
  + 7 = Rank [Descending]
* Special Action
  + 0 = Kick from Event
  + 1 = Change Score

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Procedure ID | Procedure Name | Assignment ID | Distribution ID |  |  |  |  |  |

* Assignment ID: Used to select which assignment will be populated

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Distribution ID | Assignment ID | Role ID |  |  |  |

* Assignment ID: Which Assignment the user must have
* Role ID: Which role the user might have