

# ManageHelp

## Sprint 1 Planning Document

Tom Carsello, Matt Hiatt, Sharan Sivakumar, Jon Tokad

### Sprint Overview

During the first sprint we want to create the basis of our application. We want to set up the user authentication system so that we have the start of a functional application at the end of the sprint, but our main focus will be on developing the user interface. A strong initial user interface will give us the foundation we need to add new features, connect components, and optimize our application throughout the rest of the project.

**Scrum Master:** Tom Carsello

**Meeting Times:** Tuesday 9am, Friday 8:30am (with project coordinator)

### **Risks and Challenges:**

The biggest challenge during this first spring will be rapidly gaining familiarity with the technologies in the MERN stack. While we have been trying to familiarize ourselves with the basics, it will be difficult to develop a sprint's worth of content while trying to learn the languages and the basics of web application development from almost scratch. We must make sure to not get lost researching and to ensure we are learning mostly by doing so that we have a quality artifact to present at the end of the sprint. Additionally, we will have to be flexible because even though we planned the UI feel and look out in the previous documentation, there will likely be unanticipated implementation challenges that require adaptability and some on the fly design choices to overcome.

## Current Sprint Detail

**User Story (1/12):** *As a user, I would like to be able to create and log in to my ManageHelp account*

Task #	Description	Time	Owner
1	Set up Back End Server (Node/Express)	6h	Matt
2	Set up Database (MongoDB)	5h	Matt
3	Configure authentication	3h	Matt
4	Pipe user information to database	1h	Matt
5	Set up web client template	6h	Jon
6	Set up web client and routes	4h	Jon
7	Implement UI for account creation	3h	Sharan
8	Implement UI for login	3h	Sharan
9	Implement login credential validation	3h	Matt
10	Route user to workspace selection page after login	2h	Jon
11	Notify user when username/password is invalid	1h	Jon
12	Use unit tests to debug and verify the authentication system works	4h	Jon

## Acceptance Criteria

- Given that the user's email is not already associated with an account and that their password is not too weak, the user's account will be created and added to the database
- Given that the backend user login credential validation functionality works properly, a user should be able to input a username and password and the system should verify the account details from the database.
- Given that the backend user login credential validation functionality works properly, once the user's account details are verified, the system needs to take the user to the correct home page based on their level of access
- Given that the backend user login credential validation functionality works properly, the system needs to notify the user whenever they input a wrong combination of either username or password.

**User Story (2/12):** *As an admin, I would like to be able to create and edit my Company's ManageHelp Workspace*

#	Description	Time	Owner
1	Upon workspace creation, add workspace to database and set the creating user as its admin	4h	Jon
2	Implement create workspace UI	3h	Jon
3	Allow admin to change the name and join code for their organization workspace	1h	Jon
4	Use unit tests to assess functionality of the workspace creation	4h	Jon

#### Acceptance Criteria

- Given that the backend workspace functionality is implemented properly, when a user tries to create a workspace, the workspace will be added to the database along with the user as an admin.
- Given that the admin has created their account successfully, the home screen should contain an option to view the workplace information like join code, company name, and privileged users.
- Given that the admin can properly view workspace information, they should also have the ability to edit particular fields and those fields should be updated in the database.

**User Story (3/12) :** *As an employee, I would like to be able to join my company's ManageHelp workspace*

#	Description	Time	Owner
1	Implement UI for user to input organizational join code	3h	Tom
2	Route user to the appropriate workspace	2h	Tom
3	Update workplace database with new user info	1h	Tom
4	Create unit tests to test employee joining ability	1h	Tom

#### Acceptance Criteria

- Given that a user has signed up for an employee account, they can then enter a workspace join code given to them by their boss (workplace admin.)

- Given that the user has supplied an invalid join code, the system must notify the user that the code was invalid.
- Given that a user has logged in, the system should list all workspaces they are a member of, and can click on them to access that workspace.
- Given that a user has signed up for an employee account, the workplace database should be updated with the new information of the user.

**User Story (4/12):** *As a admin, I would like to be able to remove employees from the workplace*

#	Description	Time	Owner
1	Implement managerial removal function in backend	2h	Tom
2	Create remove tab and UI in web client	2h	Tom
3	Notify employee that they have been removed via email	2h	Tom
4	Unassociate user account with that workspace in the database	1h	Tom
5	Use unit tests to debug and verify removal function works	2h	Tom

#### Acceptance Criteria

- Given that a user is an admin for the workplace, they should be able to remove employees at their discretion.
- Given that a user has been removed from a workspace, the system should notify the user via email when the manager removes them from the workspace.
- Given that the functionality to remove employees is implemented correctly, the user data for that workplace should be wiped and they should only be able to access any other workspaces they were in

**User Story (5/12):** *As a admin, I would like to invite employees via email to join a workspace*

#	Description	Time	Owner
1	Set up mailer and ability to email users	6h	Matt
2	Implement email text with link to join an existing	2h	Matt

	workplace		
3	Route email recipient to account creation page then to the appropriate workspace	1h	Matt
4	Use unit tests to test mailer works with dummy email addresses	2h	Matt

### Acceptance Criteria

- Given that a workspace has been configured correctly, the admin should be able to send an email containing a link to join the workspace to a not-yet-registered employee.
- Given the recipient's email is not yet associated with a ManageHelp account, they will be guided to the account setup page.
- Given the recipient's email is already associated with a ManageHelp account, they will simply be added to the workspace once joining via the link.

**User Story (6/12):** *As a manager, I would like to be able to assign job titles, roles, and pay rates to employees*

#	Description	Time	Owner
1	Implement managerial role assignment function	1	Tom
2	Notify employees when there has been a change in their job title, role, and pay rate	3	Tom
3	Implement managerial pay assignment function	1	Tom
4	Implement admin role assignment function	1	Tom
5	Test assignment functionality with written unit tests	2	Tom

### Acceptance Criteria

- Assuming the user is an admin account, they should be able to assign roles to various employees
- Assuming the user is a manager, they should be able to assign pay rates to various employees
- Assuming the user has signed up for a managerial account, the system should notify the user when there has been a change in their current roles/pay rates.

**User Story (7/12):** *As a manager, I would like to be able to view information about my employees in an easy to read format*

#	Description	Time	Owner
1	Retrieve all employee information from workspace database	2h	Tom
2	Implement UI to display all employees for manager to see, and enable their information to be shown if they are clicked on	2h	Tom
3	Allow employees to be listed or grouped by various attributes (pay rate, role, etc)	1h	Tom

#### Acceptance Criteria

- Given the manager selects the view employees button, they are presented with all of and only the employees in that workspace.
- When the manager views the employees, s/he should have the ability to change how they are listed/sorted.
- Given edit functionality is implemented properly, the manager should be easily able to move to the edit function from the view screen to adjust information

**User Story (8/12):** *As an employee, I would like to be able to input scheduling restrictions*

#	Description	Time	Owner
1	Implement UI for an employee to edit their scheduling restrictions	2	Jon
2	Update scheduling data in the database	1	Jon
3	Display restrictions for each employee when a manager is attempting to schedule them	2	Jon
4	Write and test functionality with unit tests	2	Jon

#### Acceptance Criteria

- Given that an employee has joined the workspace, they should be able to add times they can not work into their profile.
- Given that the input restrictions functionality is working properly, the manager should be able to see all employees restrictions in the employee view screen.

- Given that an employee has input restrictions, the scheduler should warn a manager when they are scheduling that person for their restricted times.

**User Story (9/12):** *As an admin, I would like to be able to delegate scheduling permissions to shift managers*

#	Description	Time	Owner
1	Implement UI for an admin to delegate scheduling permissions to shift managers	2	Sharan
2	Update schedule permissions in the database	1	Sharan
3	Display notifications when a shift manager's scheduling permissions are delegated	2	Sharan

#### Acceptance Criteria

- Given that the user is an admin, they should be able to delegate scheduling permissions to shift managers
- Given that the user is an admin, when they delegate schedules to shift managers, the scheduling permissions should be updated in the database
- Given that the user is an admin, when they delegate schedules to shift managers, the shift managers should be notified.

**User Story (10/12):** *As a user, I would like to be able to submit requests for days off to the manager for approval*

#	Description	Time	Owner
1	Create UI for a user to be able to submit requests for days off to the manager for approval	5	Sharan
2	Update requests for days off in the database	5	Sharan
3	Notify users when a manager approves their request for a day off	5	Sharan

#### Acceptance Criteria

- The user should have a clean and intuitive user interface to be able to submit requests for days off to the manager for approval
- When a request for a day off is made, it should be updated in the database
- When a user's day off is approved, they should be notified

**User Story (11/12):** *As a user, I would like to be able to reset my password via (email, security questions?)*

#	Description	Time	Owner
1	Create UI for password recovery screen	1hr	Sharan
2	Create password recovery system	3	Sharan
3	Update database when a user's password is changed	1	Sharan
4	Notify the user when their password is changed	1	Sharan

#### Acceptance Criteria

- Given that a user has forgotten their password, any user should be able to reset their password via email
- Given that a user has begun the process of resetting their password, there should be clear instructions and a useful UI to guide them.
- Given that a user has changed their password, the password (hashcode) for that user should be updated in the database
- Given that a user has changed their password, the system should notify the user of the change via email

**User Story (12/12):** *As an admin, I would like to be able to promote existing employees to managers*

#	Description	Time	Owner
1	Create UI for admin being able to promote existing employees to managers	2	Tom
2	Update the database when the permission of any user is changed	2	Tom
3	Notify employees when their permissions are changed	1	Tom

#### Acceptance Criterion

- Given that an admin has started to promote an existing employee, there should be a clear and intuitive user interface to guide them
- Given that a user is an admin, the user should be able to promote other users to the manager role



- Given that the user is an admin, when they promote an employee to a manager, the database should be updated
- Given that the user is an employee, notify them when their permissions are changed by admin

## Remaining Backlog

### In Progress

1. As an admin, I would like to be able to create my company's ManageHelp workspace
2. As a user, I would like to be able to login to my ManageHelp account
3. As an employee, I would like to be able to join my company's ManageHelp workspace
4. As a manager, I would like to be able to remove employees from the workplace
5. As a manger, I would like to invite employees via email to join a workspace
6. As a manager, I would like to be able to assign roles and pay rates to employees
7. As a manager, I would like to be able to view information about my employees in an easy to read format
8. As a manager, I would like to be able to input my company's weekly hours into the schedule template
9. As a manager, I would like to be able to create weekly schedules several weeks in advance
10. As an employee, I would like to be able to view any finished schedules when posted by the manager
11. As a user, I would like to have a system where I can communicate with other users via email or a chat system conveniently
12. As an employee, I would like to submit requests for certain days off
13. As an employee, I would like to be able to request other employees cover a shift and accept other employees covered shifts
14. As a manager, I would like to be able to veto employee shift trades/coverages
15. As a user, I would like to be able to access ManageHelp on both desktop and mobile platforms
16. As a manager, I would like to be able to view expected labor costs for a day or week
17. As an employee, I would like to be able to receive notifications when my schedule is updated
18. As an employee, I would like to be able to input scheduling restrictions
19. As a manager, I would like to be able to delegate scheduling permissions to shift managers
20. As a manager, I would like to be able to receive overtime warnings if an employee is overscheduled

21. As a user, I would like my password to not be stored in plaintext
22. As a user, I would like to be able to recover my password via (email, security questions?)
23. As a user, I would like to see the specific tasks that I've been assigned to do by my manager
24. As an user, I would like to view my earnings with and without tax
25. As a manager, I would like ways to post announcements
26. As an employee, I would like a way to view and be notified of announcements
27. As a user, I would like a way to easily navigate between different workspaces if I work two jobs
28. As a user, I would like to be able to check off tasks that I've completed
29. As an admin, I would like to assign managers tasks that only they can see
30. As a user, I would like the option to connect the chat feature to my email
31. As an admin, I would like to be able to promote existing employees to managers
32. As a manager, I would like to be able to release and un-release schedules I am working on
33. As an employee, I would like to be able to go back and view previous schedules up to a month ago
34. As an employee, I would like to be able to check off any tasks I've completed during a shift
35. As a manager, I would like to be able to attach specific tasks to an employee's shift
36. As a manager, I would like to be able to approve and reject employee day off requests