FamilyTask Test Project

Project Design Document

1. Introduction

WARNING

If you have been invited to take this test then please keep in mind the following.

1. This is not an opportunity to showcase creativity. You are expected to follow this design specification explicitly. You may ask for clarification, but a failure to follow the specification outlined in this document is an automatic test failure.

2. While there is an emphasis given to time needed for completion, more emphasis is given to the quality of the code produced.

3. We will run your code, and physically test it. Your code must be clean, and it must function in order to be considered for a position with Appech, or to be considered for Agency Work.

4. Once you turn in your code, no further attempts can be made for a minimum of six months from the time of your submission.

2. Expectations

Please implement the following features.

1. Assign Task To Member
2. Create New Task
3. Complete Task
4. Complete Task for Member

These features are outlined here in this document. We have also outlined the features of the member FamilyManager, but these features are already implemented. Use the Family Manager code as an example to understand our standards for code quality, and complete the assigned features above. Once again, please do not deviate from the designs outlined in this document. **We will fail all tests that implement additional features not requested.**

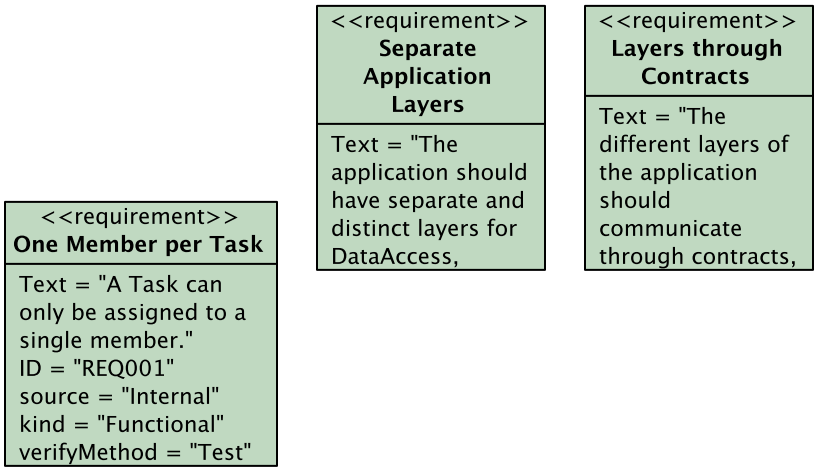
2.1. FamilyTask

2.1.1. Properties

Author	john
Company	
Description	The Family Task Application is a test application used by Appech to prove the skills of potential contractors, employees, and Development Teams. This project is built using SOLID Principals, and the software paradigms listed in this document should be followed to the letter. We evaluate code looking for the following qualities. 1. Time required to produce results. 2. Adherence to SOLID Principals in Development 3. Ability to follow directions

UML Version 2.x

2.2. WebClient Requirement Diagram



2.2.1. Layers through Contracts

ID: REQ003

The different layers of the application should communicate through contracts, and not concretions.

2.2.2. One Member per Task

ID: REQ001

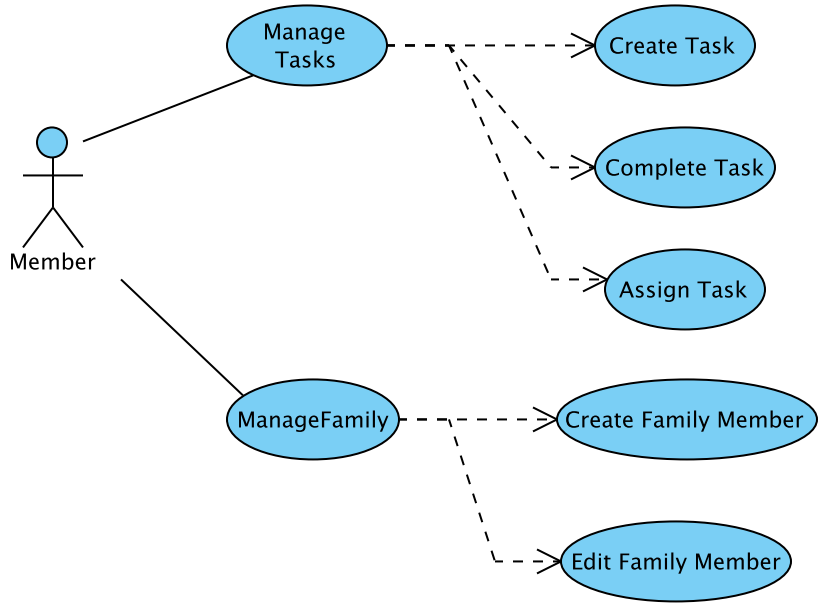
A Task can only be assigned to a single member.

2.2.3. Separate Application Layers

ID: REQ002

The application should have separate and distinct layers for DataAccess, Services, and Presentation

2.3. UseCases Use Case Diagram



2.3.1. Assign Task

ID: UC05

2.3.2. Complete Task

ID: UC04

2.3.3. Create Family Member

ID: UC06

2.3.4. Create Task

ID: UC03

2.3.5. Edit Family Member

ID: UC07

2.3.6. Manage Tasks

ID: UC01

2.3.7. ManageFamily

ID: UC02

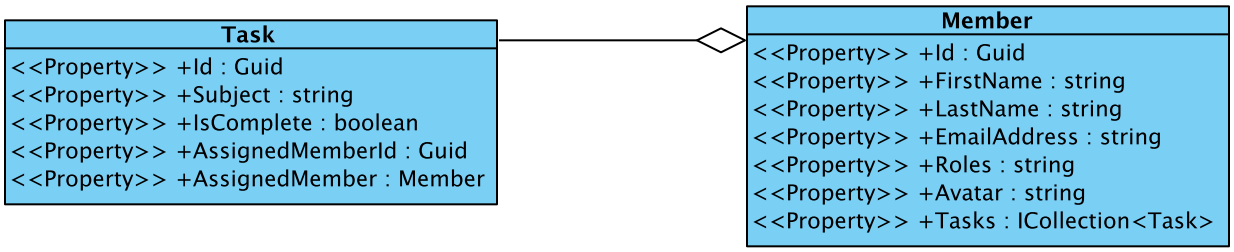
2.3.8. Member

ID: AC01

2.4. Data Dictionary

Entity Name	Entity Description					
Column Name	**Column Description**	**Data Type**	**Length**	**Primary Key**	**Nullable**	**Unique**
Member						
Avatar		integer	10	false	true	false
Email		integer	10	false	false	false
FirstName		integer	10	false	false	false
Id		integer	10	true	false	false
LastName		integer	10	false	false	false
Roles		integer	10	false	true	false
Task						
AssignedMemberId		integer	10	false	false	false
Id		integer	10	true	false	false
IsComplete		integer	10	false	false	false
Subject		integer	10	false	false	false

2.5. DataModel Class Diagram



2.5.1. Member

2.5.2. Task

2.6. Member

2.6.1. Properties

Visibility	public
Abstract	false
Leaf	false
Root	false
Active	false

2.6.2. Attributes Summary

Name	Description
Avatar	
EmailAddress	
FirstName	
Id	
LastName	
Roles	
Tasks	

2.6.3. Attributes

Avatar

Visibility	public
Type	string

EmailAddress

Visibility	public
Type	string

FirstName

Visibility	public
Type	string

Id

Visibility	public
Type	Guid

LastName

Visibility	public
Type	string

Roles

Visibility	public
Type	string

Tasks

Visibility	public

Type	ICollection<Task>

2.6.4. Relationships

Relationship	From	To
unnamed	Member	Task

2.7. Task

2.7.1. Properties

Visibility	public
Abstract	false
Leaf	false
Root	false
Active	false

2.7.2. Attributes Summary

Name	Description
AssignedMember	
AssignedMemberId	
Id	
IsComplete	
Subject	

2.7.3. Attributes

AssignedMember

Visibility	public
Type	Member

AssignedMemberId

Visibility	public
Type	Guid

Id

Visibility	public
Type	Guid

IsComplete

Visibility	public

Type	boolean

Subject

Visibility	public
Type	string

2.7.4. Relationships

Relationship	From	To
unnamed	Member	Task

3. Features In System

3.1. Family Manager

3.1.1. Create New Member

Properties

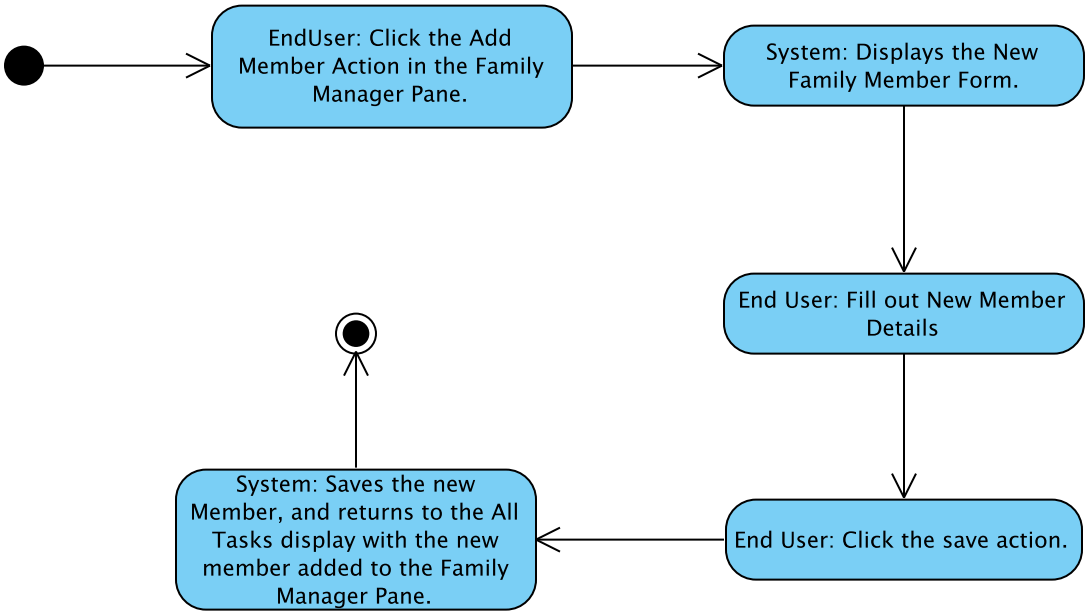
Abstract	false
Leaf	false
Root	false
Visibility	public

Children Summary

Name	Description
End User: Click the save action.	
End User: Fill out New Member Details	
EndUser: Click the Add Member Action in the Family Manager Pane.	
System: Displays the New Family Member Form.	
System: Saves the new Member, and returns to the All Tasks display with the new member added to the Family Manager Pane.	
unnamed	
unnamed	

Sub Diagrams

Create New Member Activity Diagram



- This Task is Complete Already, and can be used as an example.

3.1.2. Family Manager - UI Design

Properties

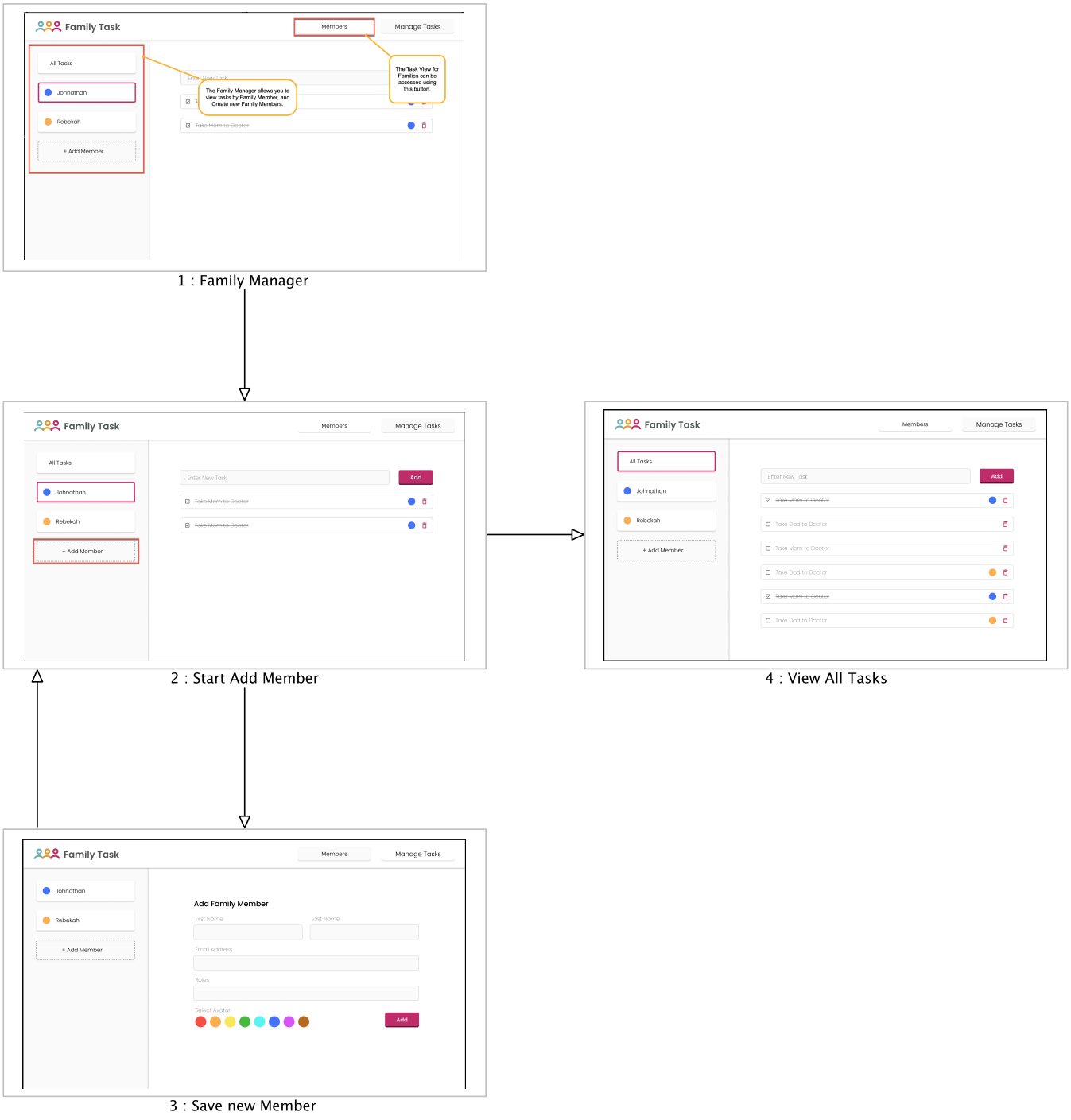
Abstract	false
Leaf	false
Root	false
Visibility	public

Children Summary

Name	Description
Family Manager	
Save new Member	
Start Add Member	
View All Tasks	

Sub Diagrams

UI Design Wireflow Diagram



3.1.3. Family Manager

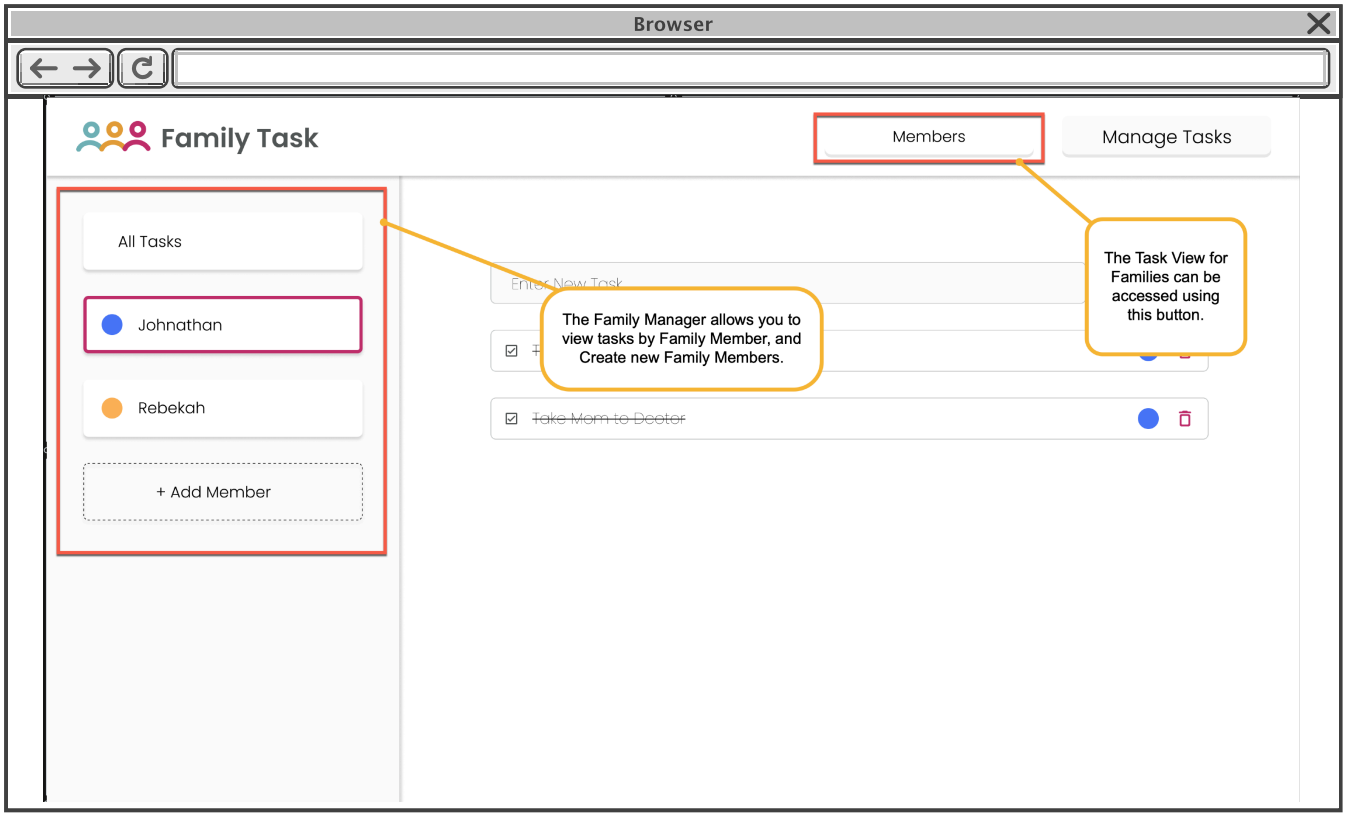
Screen ID: 1

Wireframe: Scene

Relationships

Relationship	From	To
unnamed	Family Manager	Start Add Member

Reference Diagrams



Scene - Initial

3.1.4. Save new Member

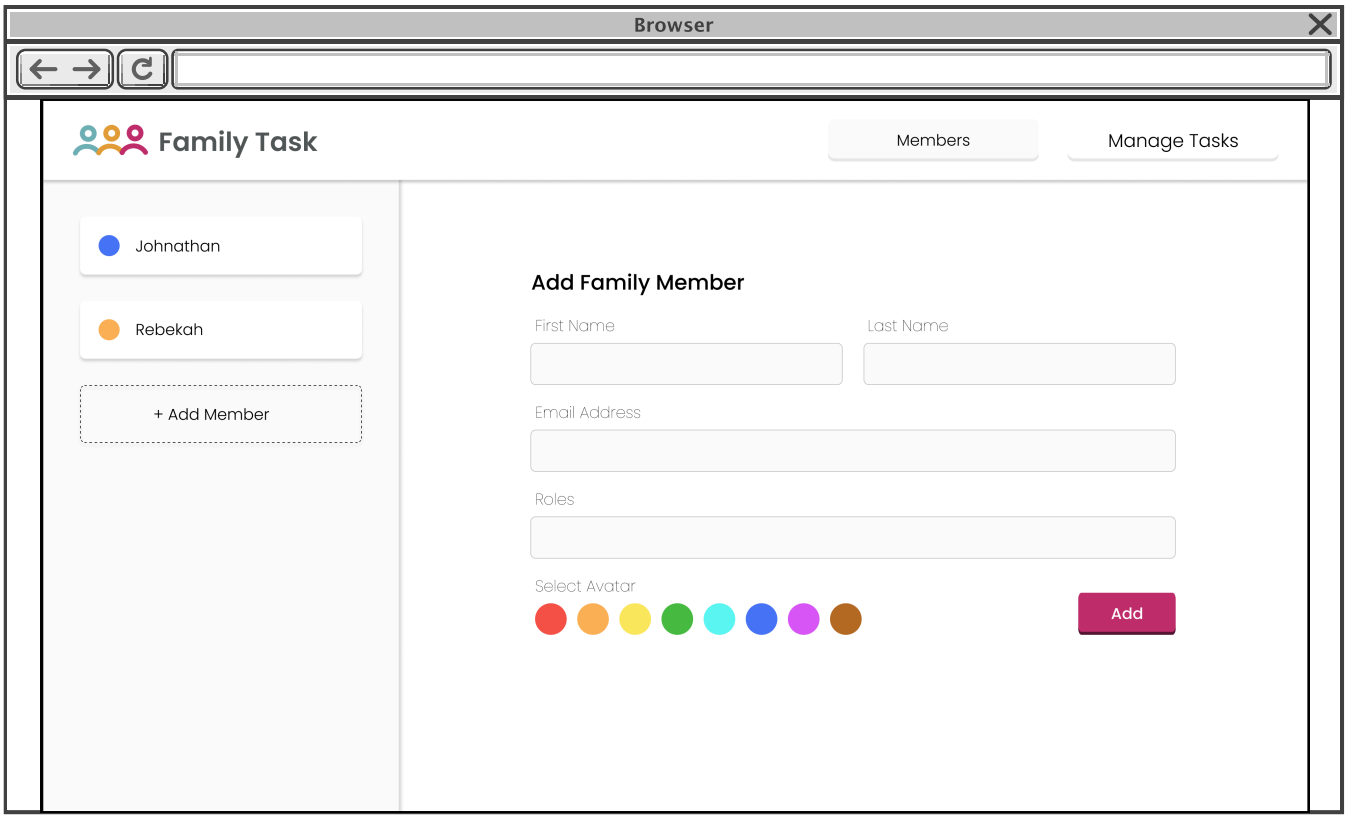
Screen ID: 3

Wireframe: Scene

Relationships

Relationship	From	To
unnamed	Save new Member	Start Add Member
unnamed	Start Add Member	Save new Member

Reference Diagrams



Scene - Initial

3.1.5. Start Add Member

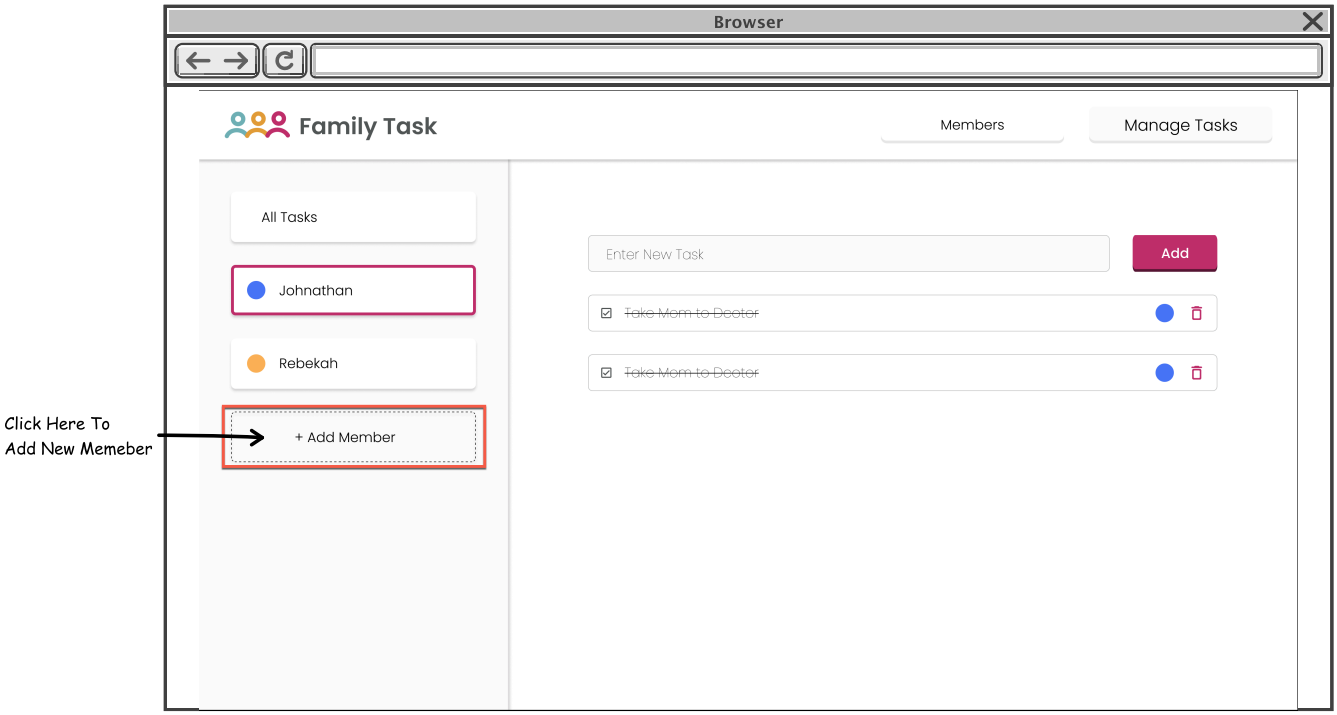
Screen ID: 2

Wireframe: Start Add Member

Relationships

Relationship	From	To
unnamed	Start Add Member	Save new Member
unnamed	Start Add Member	View All Tasks
unnamed	Family Manager	Start Add Member
unnamed	Save new Member	Start Add Member

Reference Diagrams



Start Add Member - Initial

3.1.6. View All Tasks

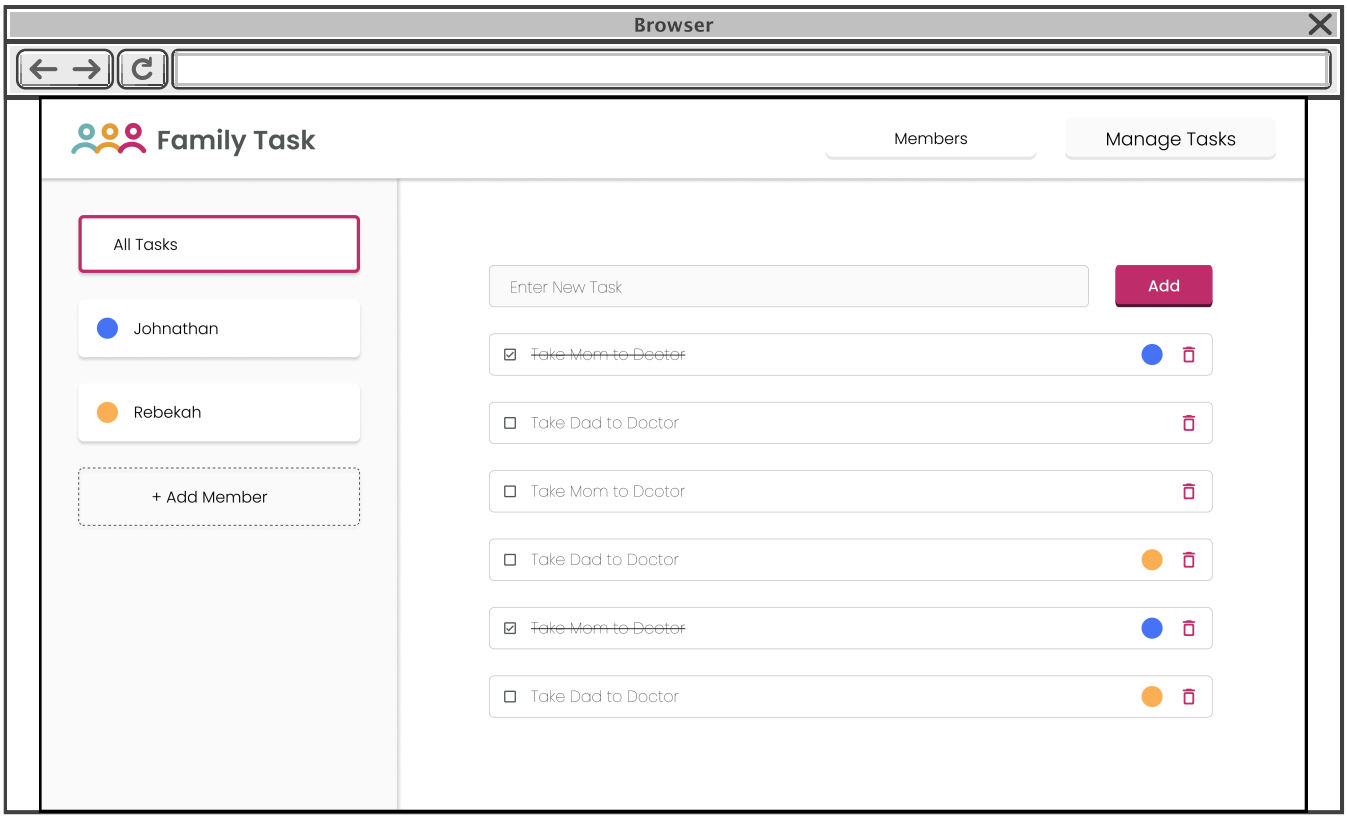
Screen ID: 4

Wireframe: View All Tasks

Relationships

Relationship	From	To
unnamed	Start Add Member	View All Tasks

Reference Diagrams



View All Tasks - Initial

The UI For this feature has already been implemented. You will only need to tie backend to the frontend.

3.1.7. View Tasks for Members

Members can be assigned tasks from the master task list. Assigning a member a task will set the Member for the Task to the Selected Member. This Story will allow the EU to view the tasks that have been assigned to the Member.

Properties

Abstract	false
Leaf	false
Root	false
Visibility	public

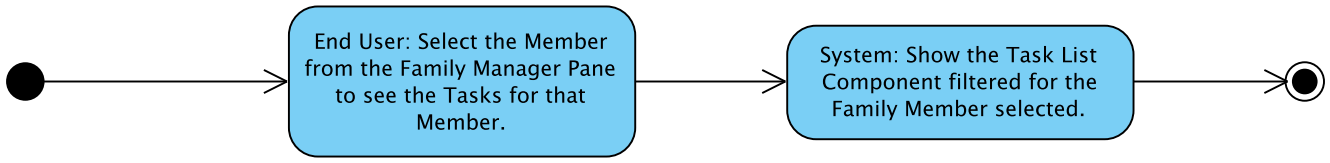
Children Summary

Name	Description
End User: Select the Member from the Family Manager Pane to see the Tasks for that Member.	

Name	Description
System: Show the Task List Component filtered for the Family Member selected.	
unnamed	
unnamed	

Sub Diagrams

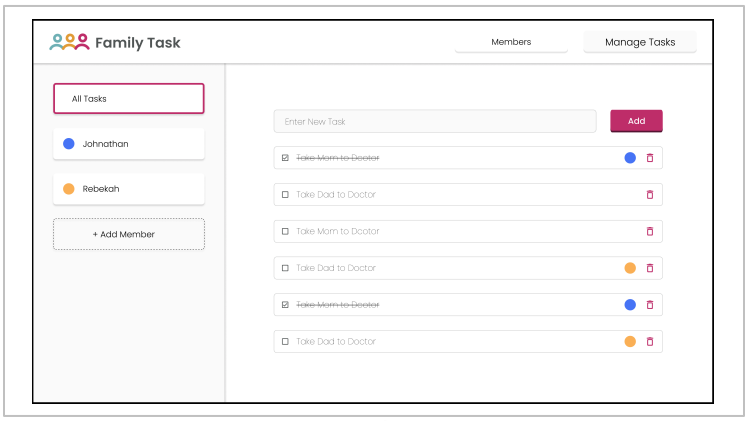
View Tasks for Member Activity Diagram



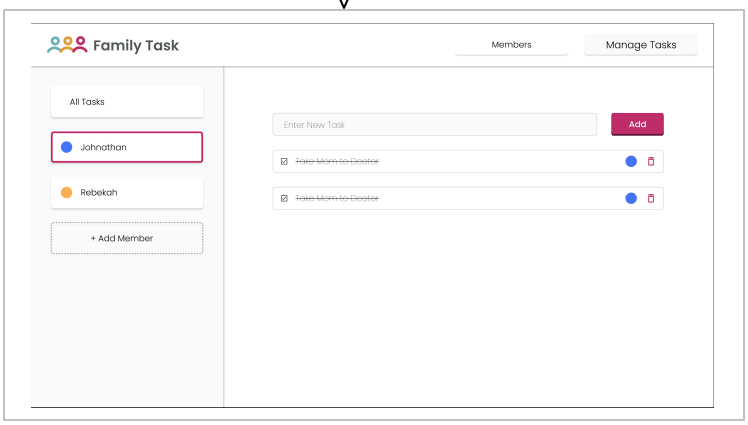
3.2. Task Manager Feature

3.2.1. UI Design

3.2.2. Task Manager - UI Design Wireflow Diagram



1 : Task List



2 : Filtered – Task List

3.2.3. Filtered - Task List

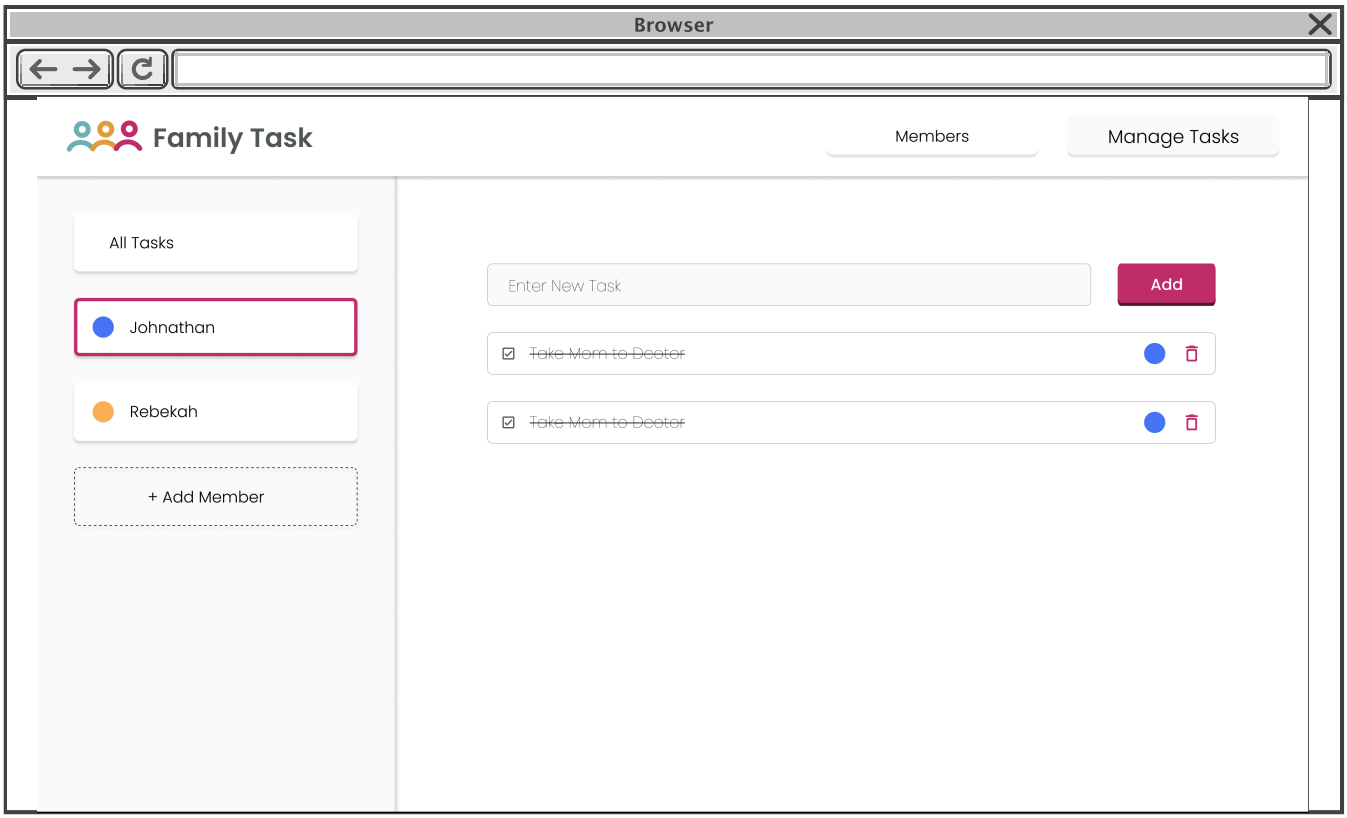
Screen ID: 2

Wireframe: Filtered - Task List

Relationships

Relationship	From	To
unnamed	Task List	Filtered - Task List

Reference Diagrams



Filtered - Task List - Initial

3.2.4. Task List

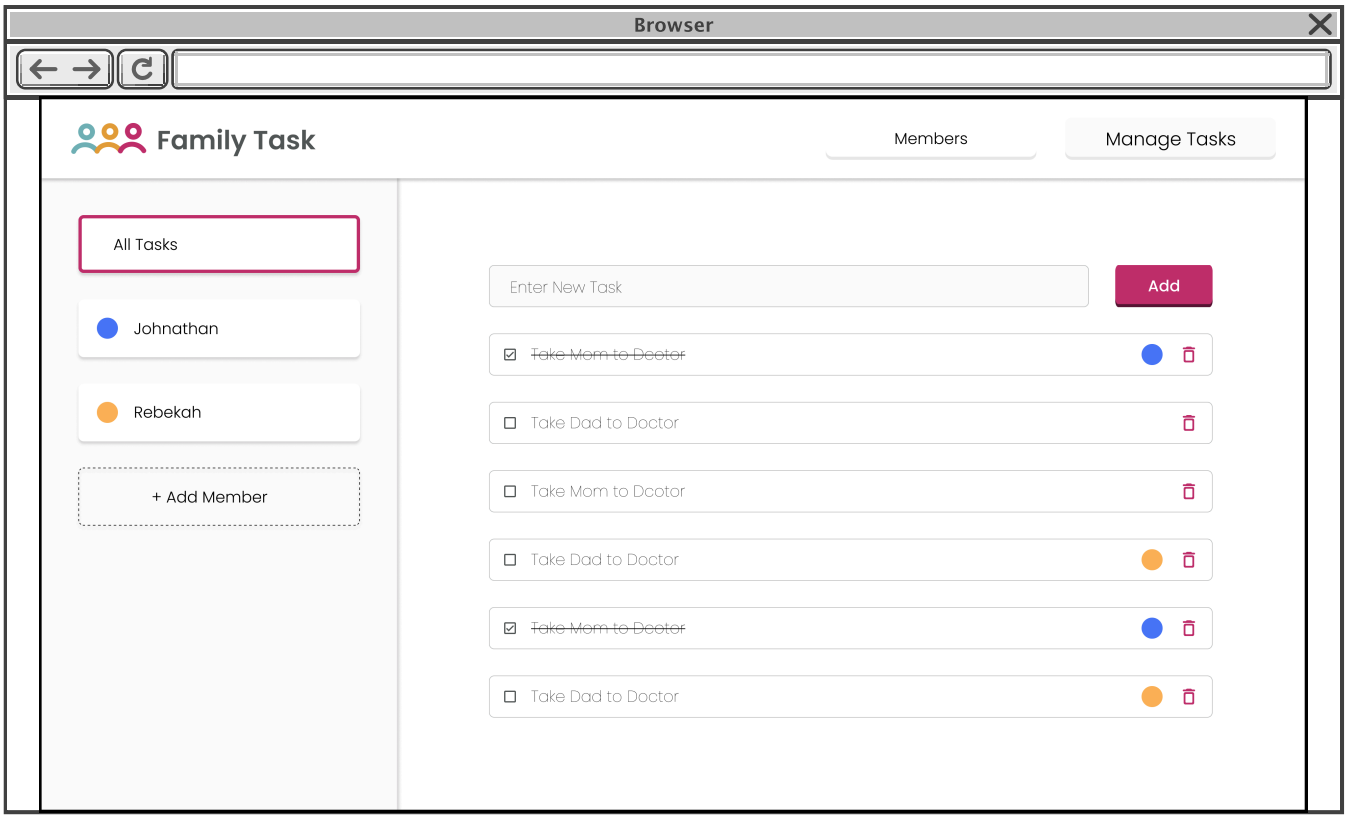
Screen ID: 1

Wireframe: Task List

Relationships

Relationship	From	To
unnamed	Task List	Filtered - Task List

Reference Diagrams

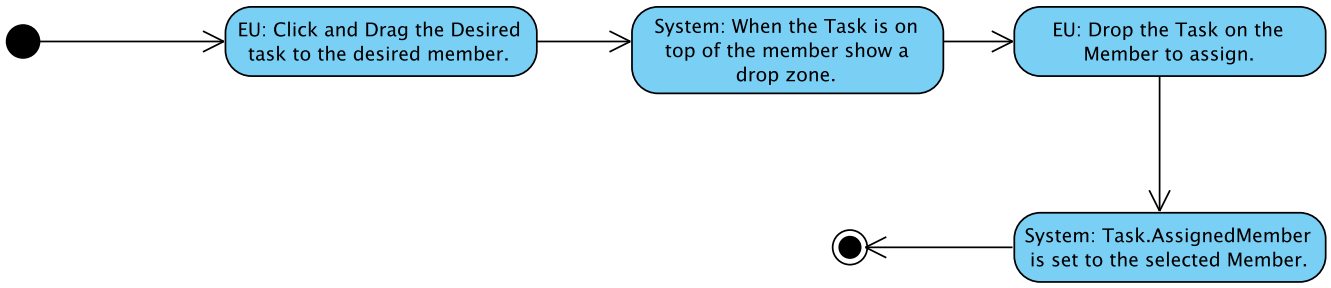


Task List - Initial

3.2.5. Assign Task To Member

Tasks in the System can be assigned to a Member by Dragging the Task to the desired member in the Family Manager View.

3.2.6. Assign Task To Member Activity Diagram



3.2.7. Complete Task

Allow the EU to mark a task complete by checking the check box for the desired Task. When this occurs the Task.IsComplete Column is set to true.

3.2.8. Create New Task

Allow the EU to Create a new Task by entering it at the top of the Task List Component. The new task should immediately show up in the list, and be saved to the database through the API.

3.2.9. Create Task for Member

When viewing the tasks for a specific member, allow the EU to create a task at the top of the Task List Component and then add that task to the database with the Assigned Member set to the currently selected member.