Calendar Component

CalendarEvent

+	id	String
+	title	String
+	date	String
+	isAllDay	boolean
+	startTime	String
+	endTime	String
+	userld	String
+	recurrence	String
+	repeatUntilDate	String
+	isRecurring(): bool	
+	occursOn(date): bool	
+	generateRecurringDa	ates(): List <string></string>

CalendarService

+	addEvent(event): Future <void></void>
+	updateEvent(event): Future <void></void>
+	deleteEvent(eventId): Future <void></void>
+	getEventsForDate(date): Stream <list<calendarevent>></list<calendarevent>

CalendarWidget (Main UI Class)

+	displayDayView(): void
+	displayWeekView(): void
+	displayMonthView(): void

Chore Component

Chore

+	id	String
+	title	String
+	userld	String
+	frequency	String
+	monthsDue	String
+	lastCompletedDate	String
+	dueMonthCarryover	String

ChoreService

+	addChore(chore): Future <void></void>
+	updateChore(chore): Future <void></void>
+	deleteChore(chore): Future <void></void>
+	getChores(): Stream <list<chore>></list<chore>

ChoreCard (Reusable UI Widget)

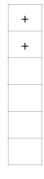
+	displayAllChores(): void
+	filterByMonth(month): void
+	toggleView(viewType): void
+	assignChoreToUser(chore, userId): void
+	unassignChore(chore): void
+	handleDrop(chore, targetUser): void

ChoreWidget (Main UI Class)

+	render(chore): Widget
+	onDragStart(): void
+	onDragComplete(): void
+	

Gro

Gro



Recipe Component

Recipe

+	id	String
+	title	String
+	Ingredients	List <string></string>

RecipeService

+	addRecipe(recipe): Future <void></void>
+	updateRecipe(recipe): Future <void></void>
+	deleteRecipe(recipe): Future <void></void>
+	getRecipes(): Stream <list<recipe>></list<recipe>

Date

+
+
+

RecipeWidget (Main UI Class)

+ displayAllRecipes(): void		displayAllRecipes(): void
	+	assignRecipeToDay(recipe, day): void
	+	unassigneRecipe(recipe): void
	+	handleDrop(recipe, targetDay): void

RecipeCard (Reusable UI Widget)

+	render(recipe): Widget
+	onDragStart(): void
+	onDragComplete(): void
+	addIngredientToGroceryList(ingredient): void

Grocery Component

String
String
bool

GroceryService

+	addGrocery(item): Future <void></void>
+	updateGrocery(item): Future <void></void>
+	deleteGrocery(item): Future <void></void>
+	getGroceries(): Stream <list<grocery>></list<grocery>

displayAllRecipes(): void
toggleComplete(): bool

Shared

getWeekRange(date): List<DateTime>

getMonthGrid(date): List<List<DateTime>>

formatDisplayTime(event): String

calendarEvents

Field	Туре	Key	Relationships
id	String	PK	
title	String		
date	String		
isAllDay	boolean		
startTime	String		
endTime	String		
userld	String	FK	-> users.userId
familyld	String	FK	-> families.familyId
recurrence	String		
repeatUntilDate	String		

recipe

Field	Туре	Key	Relationships
id	String	PK	
title	String		
ingredients	List <string></string>		
familyld	String	FK	-> families.familyld

groceryItem

Field	Туре	Key	Relationships
id	String	PK	
title	String		
isComplete	boolean		
familyld	String	FK	-> families.familyId

chore

Field	Туре	Key	Relationships
id	String	PK	
title	String		
userld	String	FK	-> users.userId
frequency	String		
monthsDue	String		
lastCompletedDate	String		
dueMonthCarryover	String		
familyld	String	FK	-> families.familyId

users

Field	Туре	Key	Relationships
userld	String	PK	
name	String		
uid?	String	FK	-> firestore users
familyld	String	FK	-> families.familyld

families

Field	Туре	Key	Relationships
familyld	String	PK	
userld	String	FK	-> users.userId