|  |
| --- |
| **TaskManager** |
| Private: |
| * taskList: ArrayList<Task> * catergoryList: ArrayList<Category> |
| Public: |
| + addTask(task: Task, category: Category): void  + deleteTask(task: Task, category: Category): void  + getTaskList(): ArrayList<Task>  + getCategoryList(): ArrayList<Category> |

|  |
| --- |
| **Category** |
| Private: |
| * categoryName: String * categoryColor: String * tasks: ArrayList<Task> |
| Public: |
| + addTask(task: Task): void  + deleteTask(task: Task): void |

|  |
| --- |
| **Task** |
| Private: |
| * title: String * description: String * dueDate: String * priority: String * completion: boolean |
| Public: |
| + markComplete(): void |

**Classes/Fields/Methods:**

1. **TaskManager class:**

* Fields: taskList(List<Task>), categoryList(List<Category>)
* Methods: addTask(Task task), editTask(Task task), deleteTask(Task task), addCategory(Category category), deleteCategory(Category category),

1. **Task class:**

* Fields: title(String), description(String), dueDate(Date), priority(String), completion(boolean)
* Methods: markComplete()

1. **Category class:**

* Fields: categoryColor(String), categoryName(String), tasks(List<Task>)
* Methods: addTask(Task task), getTasks(), deleteTask(Task task)

**Relationships:**

1. **TaskManager to Task:** composition relationship – TaskManager manages list of Task objects
2. **TaskManager to Category:** composition relationship – TaskManager manages list of Category objects
3. **Task to Category:** each Task belongs to a Category