

# Design Document (Term Project)

## ToDo List

Taylor Wacker and Kameron Weaver  
November 18, 2012

CSCI 498 Android Development, Fall 2012  
Colorado School of Mines

## **Table of Contents**

Definition	3
Description	3
User Stories	4
Use Cases	5
UML Diagram	6
Use Cases	
Create ToDo Item	7
View ToDo Item	8
Edit ToDo Item	9
Domain Analysis	10
Screen Specs	11

## **Definition**

Taylor Wacker and Kameron Weaver will be creating a ToDo list application that interfaces with the phones built in services and pre-existing software.

## **Description**

This application is designed to help keep better track of your life. Everything revolves around the ToDo item. Each item has a title and description, as well as the option to include many other features from audio recording to photo to calendar event and alarm. The most important element in a ToDo item is that it can have a list of ToDo items contained within it which gives it a tree like structure for organization. This way the user can break each big task into smaller more manageable pieces. Additional features the ToDo item can have include: recurring ToDo item, including the interval; picture; audio recording; alarm; linking to the phones calendar. Once each item is complete it can be marked as resolved.

## User Stories

Tyler the college student is continuously forgetting what all he needs to get done. Fortunately he heard of this nifty new app that will help keep track of all that he needs. He downloads the app and immediately remembers that he has a homework assignment due tomorrow in Bakos's Android class. He starts up his new app and creates a new ToDo item called homework and adds a quick description saying that he needs to finish the user stories assignments.

Later that day he remembers that he also has an econ assignment. He turns to his phone and goes to create a new ToDo. He sees that this homework thing might be contagious and far more common than just a simple task. He creates his new ToDo before he forgets. Then he adds another ToDo this one just labeled Homework. Then he goes to his other notes. Selects the move button and sets the parent to he Homework ToDo. Tyler does the same with his econ homework assignment. Now when he looks at the homework ToDo he sees that there are 2 sub assignments. After class Tyler finds his friend Cameron who invites him for a beer at a new bar he found. Tyler adds a new ToDo. In the options there is a location note. Putting this in he can now look at google maps to see where he needs to go.

After finishing his homework he goes to the assignment and checks complete. Now when he opens his list the assignment doesn't show up. A few minutes later he wants to confirm he checked off the right assignment. Adjusting the settings he sets up the app to show finished assignments in grey.

Late that night as he's getting ready to go home he remembers that he has to grab a bunch of things the next morning before he goes to work. He adds yet another ToDo and adjust the settings to set off an alarm at 6:30 the next morning.

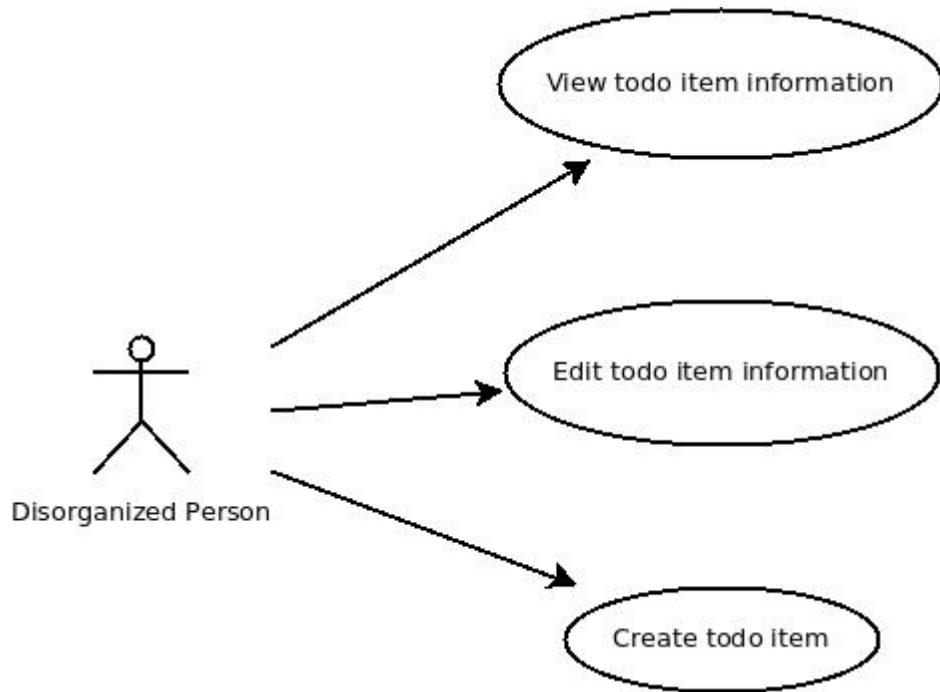
The next morning as Tyler is just getting out the door he realizes he never called his girlfriend the night before. He adds a ToDo titled "Girlfriend" with the description "Apologize profusely". Three weeks later Tyler's girlfriend has finally forgiven him. He realizes that he doesn't need the old note anymore. He goes to it and deletes it. Wanting to clean up some other assignments he looks through the settings and adjust them to delete old assignments after a day old of the completed state.

Now his life is in order. He knows what he needs to do and when he needs to do it. Thanks to the fancy schmancy new ToDo app.

## Use Cases

1. Create ToDo item
2. View ToDo item information
3. Edit ToDo item information
4. View ToDo items list
5. Flag item as resolved
6. Delete old items
7. Move ToDo item to new parent
8. Save to phone
9. Attach alarm to ToDo item
10. Attach picture to ToDo item
11. Attach sound/audio recording to ToDo item
12. Attach ToDo item to calendar
13. Attach time estimate
14. Attach location to ToDo item
15. Attach a recurring interval to an item
16. Attach a priority to an item
17. Share a list with another user

## UML



## **Use Case 1: Create a ToDo Item**

### **Main Success Scenario**

In order to benefit from the ToDo app a disorganized person needs to be able to create more ToDo items. The disorganized person selects to add a new item and is directed to a basic form screen where he/she inputs the information for this new ToDo. Upon clicking save the item is in the system for viewing.

## Use Case 2: View ToDo Item

<b>Title</b>	As a disorganized individual I want to see what I have to do so that I don't forget to do it.
<b>Primary Actor</b>	Disorganized Individual
<b>Stakeholders &amp; Interest</b>	Disorganized Individual wants to see what he/she has to do
<b>Preconditions</b>	A. Disorganized Individual accesses the app or B. Disorganized Individual selects a ToDo item
<b>Postconditions</b>	A ToDo item appears on the device screen
<b>Main Success Scenario</b>	1. Disorganized Individual sees a ToDo item
<b>Extensions</b>	1.1. The ToDo has no child items 1.1.1. Message "no child ToDos" appears
<b>Special Requirements</b>	This is the default view of the app and appears on startup.
<b>Technology &amp; Variation List</b>	Uses device built in database using SQLite.
<b>Frequency</b>	Multiple times per session: at startup and every time a new ToDo item is selected to be viewed
<b>Open Issues</b>	What should be done for corrupted information?

## **Use Case 3: Edit ToDo Item**

### **Main Success Scenario:**

In order to keep an accurate, consistent list of ToDo items, a disorganized individual should be able to edit the information of any single ToDo item. When viewing their items, a user should be able to edit the item's information. If the option to edit the item is selected, then the disorganized individual will see a screen with a form containing all of the information tied to the ToDo item. The disorganized individual can then change any of the fields (title, description, due date, audio file, picture file, priority, etc.) and select a "Save" button which will update the database information for the said item.

### **Alternate Scenarios:**

#### **Item Information Invalid:**

If the disorganized individual inputs invalid information into any of the edit fields, then they will see a prompt explaining the issue(s). This will indicate which field is invalid and inform the user that the information could not be saved due to this issue.

#### **User Cancels Edit Request:**

The disorganized individual could, anytime during the process of editing the information, decide they wanted to cancel the request to edit the information. This will scrap any changes to the data the user began to make, and return them to the ToDo list.

#### **ToDo Item Information Cannot Be Saved Due To Technical Issue:**

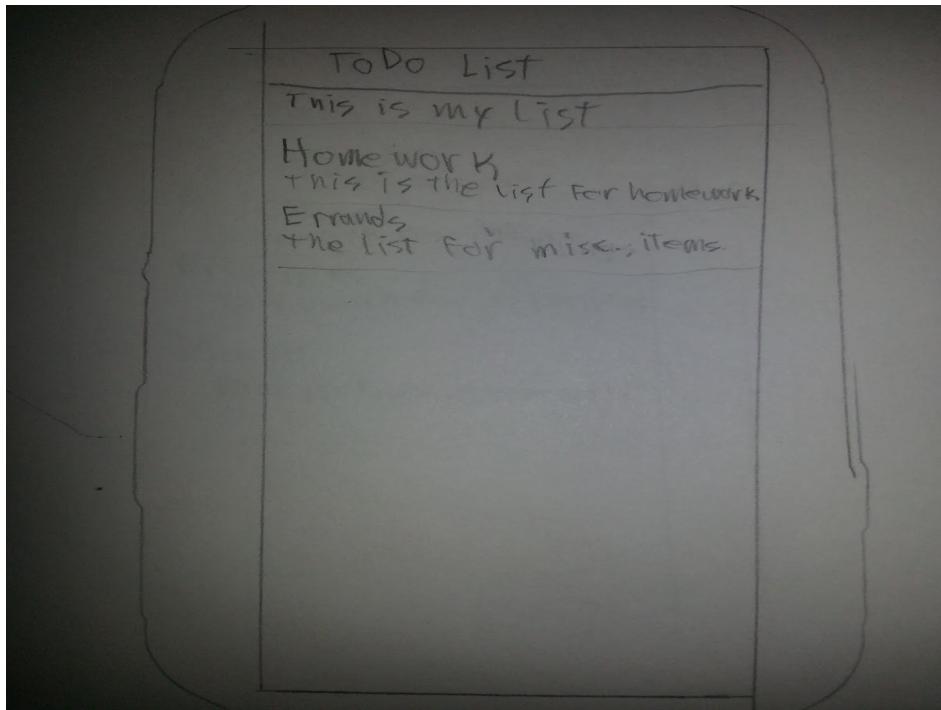
If the disorganized individual selects the save button and the application runs into an issue when trying to update the information in the database, the disorganized individual will be notified that the information could not be saved and to try again later.

## **Define Domain**

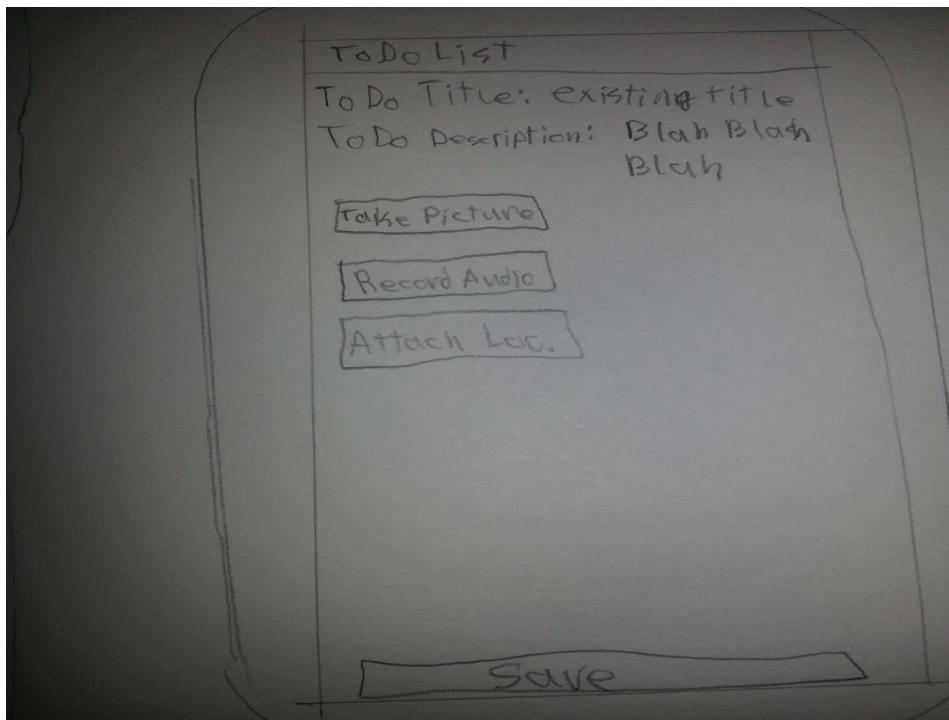
ToDo Item - An item that contains a title, description, status, priority, picture, sound recording, location, time estimate, and a list of more ToDo Items beneath it. There are many helpers that will interact with the various features that get added to the ToDo item however no other non-trivial data members existent within this project.

## **Screen Specification**

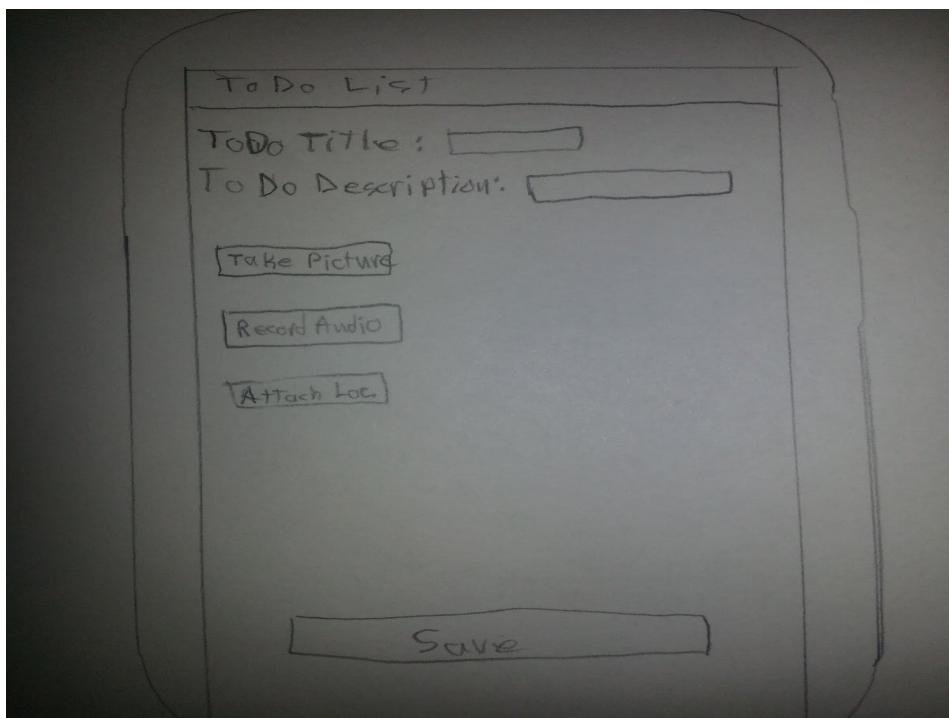
Home screen



Edit ToDo Item



Create ToDo Item



View ToDo Item

