# Functional Requirements

1. **Stakeholders**
   1. Everyone who want to have an efficiency calendar to manager their time, and find best way to finish their tasks
2. **Actors and Goals**
   1. People who want to easy manager their time
   2. People who want finish their tasks efficiently
3. **Casual Use Case Descriptions**
   1. TimeSprite will response the user’s action from touch screen. The users will tap buttons on main screen, and application will go into screen which user’s choice. The main screen will be the first stage, and it will include but no limited those stages: Setup Profiles, Add Tasks, and Generate Calendar.
   2. For edit and delete task, those two use case will only be touched off on task details stage.
4. **Fully-Dressed Use Case Descriptions**

**Pre-Condition for all use cases: iPhone or iPad is on→ TimeSprite is open to the Home Stage**

* Use Case: Setup profiles
  1. This use case will be touched off only at the first time when TimeSprite running!

1. User enters unavailable time for each day
2. User enters relax time for each day
3. User taps finished button → System: check input is legal → pull events and tasks from system calendar → set busy time → save unavailable time, relax time, and busy time to file

* Use Case: Add Tasks

1. User tap Add Tasks button
   1. Add Tasks stage show up
2. User enters task information
3. User tap save button
   1. System check input is legal
      1. Illegal input → display error message
         1. User tap OK, and change information, go back to 3. User tap save button case
      2. Input acceptable
         1. Save task information to file
         2. Pop up message ask user if want to add more tasks
            1. If user want to add more tasks, go back to 3. User tap save button case
            2. If user doesn’t want to add more tasks, go back to main stage

* Use Case: Generate Calendar

1. User tap generate calendar button
   1. “Generating….” Message show up
   2. System read user profile
      1. System read unavailable time, relax time, and busy time from profile
   3. system read tasks files
      1. system read task information from tasks files
   4. check task can be divided
      1. If task can be divided, then divided task to small parts in order can fill up free time. → assign small tasks to each free time space.
      2. If task cannot be divided, find enough free time to assign task
   5. Save each tasks to system calendar
   6. Read events from system calendar, and show calendar.

* Use Case: Task details

1. User tap task in calendar stage
2. Show task details.

* Use Case in Task details stage
* Use Case: Edit Task(This use case will use system calendar app)

1. User tap Edit Task button →System read events from system calendar → System show calendar to stage
   1. User tap event to edit →system show edit stage
   2. User change event information
   3. User click save button → System check input is legal
      1. Illegal input → display error message
         1. User tap OK, and change information, go back to 3. User tap save button case
      2. Input acceptable
         1. Save task information to file
   4. Back to main stage

* Use Case: Delete Task(This use case will use system calendar app)

1. User click delete button → System pop up confirm message
   1. Yes → Delete task → back to show calendar stage
   2. No →keep this task → back to task details stage