

---

**Hug a Cactus Inc.**

---



**FreeTime  
Vision**

**Version 2.0**

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

## ReVision document History

Date	Version	Description	Author
22/February/2006	1.0	Initial Vision	MC
1/March/2006	1.1	Updated Vision	MC
12/April/2006	2.0	2 <sup>nd</sup> Iteration Vision	Brian Johnson

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

# Vision

## 1. Introduction

The purpose of this document is to collect, analyze, and define high-level needs and features of the FreeTime. It focuses on the capabilities needed by the stakeholders, and the target users, and why these needs exist. The details of how FreeTime fulfils these needs are detailed in the use-case and supplementary specifications.

This document will discuss the different implementations for FreeTime as well as examine its customer base. FreeTime assumes a busy workload of a client in a development process where certain specified tasks must be completed. This organizational tool is based on the techniques presented in David Allen's book "Getting Things Done: The Art of Stress Free Productivity."

## 2. Positioning

### 2.1 Problem Statement

The problem of	Successfully managing one's time and remaining organized
affects	Average computer users that have a busy life
the impact of which is	Missed deadlines, low productivity, and high stress
a successful solution would be	Implementing a system to assist users with organization and time management

### 2.2 Product Position Statement

For	Average computer users that want to organize their workload
Who	Have many tasks to complete
FreeTime	Is an organizational masterpiece
That	Assists in organizing, tracking and automating a process until completion
Unlike	Ordinary calendar programs (such as Microsoft Outlook)
Our product	Follows David Allen's GTD model

## 3. Stakeholder and User Descriptions

### 3.1 Stakeholder Summary

Name	Represents	Role
Dr. Decker	The primary customer in this process.	His changes and adaptations over time will be taken very seriously and implemented in FreeTime in the most efficient and effective manner.

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

Bake's Wife	Integrated company family values and future user of the product	Will provide input about required features and easy of use
-------------	---	--

### 3.2 User Summary

Name	Description	Stakeholder
Average User	Will have a primary use for FreeTime to aid in organization and time management.	Bake's Wife
Disabled	Has special accessibility needs for FreeTime	None

### 3.3 User Environment

FreeTime will be easily adaptable to any work style in any work environment. Whether the project has millions of dollars of assets and hundreds of people working on it, or just a few people working for a few days, FreeTime will be an asset in scheduling, tracking and reporting on any project for which it is implemented.

FreeTime's flexibility will allow deployment on a wide variety of platforms, including PDA's and some mobile phones.

### 3.4 Key Stakeholder / User Needs

Some of the existing time management software has some key problems that make getting things done more of a hassle than anything else. It takes too much time to update the daily information. These programs act more as a calendar than anything else. FreeTime will overcome this obstacle by implementing David Allen's GTD model with a simple and easy-to-use interface.

The disabled have often been invited to use regular programs with the assistance of a text reader or some sort of other interface. FreeTime will incorporate the most accessible version of the new handicap aiding software. Any person with a computer will be able to use this product with no difficulty whatsoever.

Need	Priority	Concerns	Current Solution	Proposed Solutions
Manage time	Medium	Missed deadlines and appointments	Calendars, day planners, PDA's	Support for due dates, and reminders of approaching deadlines
Organize tasks	High	Remembering everything, and prioritizing it	Post-It notes or other paper system	Simple user interface that facilitates the GTD model

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

## 4. Product Overview

### 4.1 Product Perspective

FreeTime has the ability to adapt to any work center, any management style and any size project. Any project big or small can be handled by FreeTime. The user will input task information into FreeTime, which will be processed according to the GTD model. FreeTime's automation will generate electronic reminders (such as e-mails) that can be sent to users. Data can be generated and sent to cell phones or other personal PDA's in the workplace.

### 4.2 Summary of Capabilities

FreeTime

Customer Benefit	Supporting Features
Help for the disabled	Users with disabilities will have more convenience due to the accessibility features of FreeTime.
Any load	The reminders and automatically generated e-mail will keep any project on track for beginning to end. Any size, any length of time and any amount of people is no problem for FreeTime.
Convenience	For maximum productivity, FreeTime will adapt parts of its interface to better handle the user's pattern of activity.
Special hints	FreeTime will exploit past user activity to provide hints to assist user in deciding how to handle new stuff.

### 4.3 Assumptions and Dependencies

We are assuming the user will have a basic knowledge of e-mail generation and how to start and load a program. This will closely resemble any standard calendar program. If the user has no clue on how to run any type of business software a tutorial could easily be added in production. FreeTime could utilize pop-up notes for new features or if users have questions regarding a specific button.

FreeTime will be able to run on any platform with the necessary libraries installed. The majority of platforms on the market today, including PDAs and some mobile phones, have necessary libraries available.

## 5. Product Features

### 5.1 Hierarchal Project and Action Structure

Scheduled tasks are to be divided in a hierarchal fashion, holding smaller projects or atomic actions. This structure allows the user to quickly and easily find progress in any level of a project.

### 5.2 Progress Tracking

Project completeness, deadlines, and goals met may be reported on, keeping the user informed of productivity. Integration of these reports may be transferred over to another user to assist planning on a management level.

### 5.3 Context Filtering and Sorting

Particularly busy schedules call for several effective ways to organize and view tasks. FreeTime will exploit contexts such as home, school, or work, as an additional means of viewing and managing tasks.

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

#### 5.4 Context-Informed Priorities

Just because few contexts are generally needed for the average user does not take away from the importance of context in assessing priorities. FreeTime will exploit context to help the user to better decide on priorities.

### 6. Constraints

Time is the biggest constraint for this project. Money will not be an issue since all developer time will be donated.

### 7. Precedence and Priority

The priority of the above features is as follows. This ordering facilitates building an effective stand-alone planner and then adding additional functionality.

1. Hierarchal Project and Action Structure
2. Progress Tracking
3. Context Filtering and Sorting
4. Context-Informed Priorities

### 8. Other Product Requirements

#### 8.1 Applicable Standards

FreeTime will be developed with platform independence in mind. Therefore, it will need only to comply with the standards of the toolkits and languages being used.

#### 8.2 System Requirements

Because FreeTime is platform-independent, it will require certain libraries to be installed on the end-user's system. These libraries are available for a wide variety of platforms and are easy to install.

#### 8.3 Performance Requirements

The main performance requirement for FreeTime is reliability. An organizational tool such as FreeTime is useless if it is not reliable. If users are not able to access all of the data they have entered into FreeTime, they will most likely turn to a competitor's product.

### 9. Documentation Requirements

A user manual will be provided with FreeTime which will include a quick start guide, frequently asked questions (FAQ's), and troubleshooting tips. If necessary, tutorials will be developed to help users learn about FreeTime's functionality. An installation guide will also be developed for at least one platform.

### 10. RISKS

#### 10.1 R1 – Team cannot finish project

##### 10.1.1 Magnitude

High damage.

##### 10.1.2 Description

A challenge unique to us is that it is tempting to keep up to the level of other groups in the course while at the same

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

time realizing that we have 40% less manpower than any other group. We could overestimate what could be accomplished in a reasonable amount of time.

### *10.1.3 Impacts*

Given the weight of the project in the course, the impact could be bad grades for each of us. Since some of us may want to include the project in a portfolio when seeking future employment, not finishing would be especially bad.

### *10.1.4 Indicators*

If any of us is performing unusually poorly in other commitments, or we cannot meet deadlines, we might indeed risk not finishing.

### *10.1.5 Mitigation Strategy*

We need to estimate well. More realistically, especially given our inexperience, we need to frequently update estimated and actual times, in order to improve our ability to estimate. Another important group of estimates is available time.

### *10.1.6 Contingency Plan*

By realizing the costs and benefits of features, we can omit certain features if necessary and so better adjust to signs of not finishing the project.

## **10.2 R2 – Team devotes insufficient time to the project**

### *10.2.1 Magnitude*

High damage.

### *10.2.2 Description*

All team members have outside commitments of other classes and work. Exams will disrupt our typical work on the project and will be especially serious as finals near.

### *10.2.3 Impacts*

The impact could be severe during the stress of final exams. Group morale could suffer.

### *10.2.4 Indicators*

If a member is not keeping his commitments to the project, he is likely not spending enough time.

### *10.2.5 Mitigation Strategy*

We need to scrutinize timelines and be open with each other if we foresee that time commitments cannot be met. Although others can make up for a member's occasional lapse, we need to be especially vigilant given the reduced total free time of our smaller group size.

### *10.2.6 Contingency Plan*

A time crunch, especially around finals, is almost certain, so putting in extra time beforehand would help us better adjust to the crunch.

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

### **10.3 R3 – Development slows due to knowledge gaps**

#### *10.3.1 Magnitude*

Medium damage.

#### *10.3.2 Description*

Effective applications are often the result of applying good knowledge and experience of application frameworks. The group on the whole is relatively inexperienced in this area, particularly in graphical user interfaces and certain issues could stymie us occasionally.

#### *10.3.3 Impacts*

Slowing development could reduce the amount of functionality we plan. It could force us to try to incorporate functionality in the form of already completed libraries that may not fit as well as what we might develop ourselves if our development were quicker.

#### *10.3.4 Indicators*

If deadlines are not met or we are spending excessive time learning bits of the application framework, we could be sure that development is not keeping pace.

#### *10.3.5 Mitigation Strategy*

We will undoubtedly need to learn new technologies in order to develop this application. We should exploit the helpful example code that abounds on the internet. Moreover, excellent tutorials are readily available for the technologies we are considering.

#### *10.3.6 Contingency Plan*

If development slows, we need to readjust the workloads to better match the members' capabilities. Any particular deficiencies should be made clear to all.

### **10.4 R4 – Testing is inadequate**

#### *10.4.1 Magnitude*

Medium risk.

#### *10.4.2 Description*

Not testing enough is a likely risk; our inexperience likely results in overconfidence. Moreover, choosing a dynamically typed language would seem to indicate the need for more testing than we are used to doing.

#### *10.4.3 Impacts*

Insufficient testing could result in bad demonstrations, which would hurt our grades. By not finding bugs earlier through testing, our development effort could be greatly hurt later.

#### *10.4.4 Indicators*

If bugs are unusually hard to find and demos are unreliable, it is likely that more testing needs to be done.



FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

#### 10.4.5 Mitigation Strategy

We need to discuss testing in order to remind each other of its immediate relevance to our implementation, particularly given the agile context. Recording testing results will be good encouragement.

#### 10.4.6 Contingency Plan

If we are not testing enough, the Support Manager should more actively encourage the others, perhaps by developing scripts to ease testing.

## 11. Glossary

### Actionable

Whether an action step or a project can be found for an item, as opposed to finding that one can eliminate it, incubate it, or file it for reference

### Context

A set of factors such as location or tools at hand that limit what you can do at the moment

### Defer

How to handle a next action if it cannot be done or delegated. It becomes a list of next actions to track and do later.

### Delegate

How to handle a next action if a more appropriate person should do it

### Do

How to handle a next action if less than two minutes are needed to finish

### Eliminate

Get rid of an unneeded item

### GTD

A way to increase personal productivity, as described in the book "Getting Things Done" by David Allen

### Inbox

Any way in which stuff is collected for further organizing

### Incubate

File away an item that may need action later but none now

### Item

A single piece of "stuff"

### Next Action

The next physical, perceptible activity to do to bring an item closer to a wanted result. It can be done, delegated, or deferred.

### PDA

Personal Digital Assistant (PDA) is a handheld device that originally served as an organizational tool, but has gradually acquired additional possible functions, including mobile phone, media player, and web browser

FreeTime	Version: 2.0
Vision	Date: 12/April/2006
2 <sup>nd</sup> Iteration Vision	

**Project**

Any wanted result needing more than a single action step

**Project Plan**

For a project, the result of defining the purpose and principles, imagining an outcome, brainstorming, organizing, and then identifying the next actions

**Rails**

A web application framework written in Ruby

**Reference**

An item that is perhaps useful information needed for something later

**Ruby**

An object-oriented programming language with first-class and higher order functions; dynamic typing and many other features from popular object-oriented and scripting languages

**Stuff**

Anything in your life that as yet has no proper place but for which you have yet to find the wanted result or next action step

**User**

Someone who uses FreeTime for getting things done

**Waiting For**

A list of reminders of things expected to get done by others for you