### **Automatic Time Management System**

(ATOMS)

# **CS 3337 Software Engineering**

# **Application Requirements Specification Document**

## Prepared By:

LaFrance, Montague

Chan, Micky

Gomez, Carlos

Vargason, Austin

# **T**able of Contents

|  |  |  |
| --- | --- | --- |
| Section Number | Section Name | Page Number |
| 0.0 | Document Revision History | [3](#_Document_Revision_History) |
| 1.0 | Introduction |  |
| 1.1 | Purpose |  |
| 1.2 | Intended Audience and Reading Suggestions | [4](#_1.2_High_Level) |
| 1.3 | Product Scope |  |
| 1.4 | Definitions, Acronyms, and Abbreviations |  |
| 1.5 | References |  |
| 2.0 | Overall Description |  |
| 2.1 | Product Perspective |  |
| 2.2 | Product Functions |  |
| 2.3 | User Classes and Characteristics |  |
| 2.4 | Operating Environment |  |
| 2.5 | Design and Implementation Restraints |  |
| 2.6 | User Documentation |  |
| 2.7 | Assumptions and Dependencies |  |
| 2.8 | Apportioning of Requirements |  |
| 3.0 | External Interface Requirements |  |
| 3.1 | User Interfaces |  |
| 3.2 | Hardware Interfaces |  |
| 3.3 | Software Interfaces |  |
| 3.4 | Communications Interfaces |  |
| 4.0 | Requirements Specification |  |
| 4.1 | Functional Requirements |  |
| 4.2 | External Interface Requirements |  |
| 4.3 | Logical Database Requirements |  |
| 4.4 | Design Constraints |  |
| 5.0 | Other Nonfunctional Requirements |  |
| 5.1 | Performance Requirements |  |
| 5.2 | Safety Requirements |  |
| 5.3 | Security Requirements |  |
| 5.4 | Software Quality Attributes |  |
| 5.5 | Business Rules |  |
| 6.0 | Other Requirements |  |
|  |  |  |

# Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version # | Revision Author | Revision Summary | Revision Date |
| 0.1 | Austin Vargason | Initial Draft | 2/23/19 |
| 0.2 | Austin Vargason | Meeting Template Requirements | 3/9/19 |

# Introduction

This document seeks to outline the software requirements for the Automatic Time Management System (ATOMS) android application. ATOMS is an Android Application that helps the user automatically manage their time. The ATOMS application allows users to schedule projects, assignments, goals, or other activities around their current schedule. ATOMS allow users to integrate their existing Google Calendar data, which creates an intuitive scheduling experience. The goal of the ATOMS development team is to improve time management and reduce stress for our users.

## 1.1 Purpose

This SRS document version 0.1 contains specifics for the ATOMS application for Android. In this document, the overall description of the product, as well as the product’s functional, interface, design, and, security, and safety requirements will be outlined. The architects involved in the completion of the ATOMS project shall use this document to create an overall Solution Design Document (SDD).

## 1.2 Intended Audience and Reading Suggestion

This document is intended to be viewed by developers and architects involved with the ATOMS application to fully understand the requirements of the application and the interfaces it presents. This document may also be viewed by third-parties under the scope of product presentations.

## 1.3 Product Scope

The scope of the ATOMS application is Android Users on operating systems at or above: Android version 4.0.3 Ice Cream Sandwich

## 1.4 Definitions and Acronyms

Android Studio: Development environment to develop Android Applications

Java: Object Oriented Programming Language

GUI: Graphical User Interface

ATOMS: Automatic Time Management System

API: Application Programming Interface

OS: Operating System

XML: extensible markup language

Strictly Scheduled Event: A calendar event that has a pre-defined start and end time

Automatically Scheduled Event: A calendar event scheduled by the defined ATOMS algorithm

## 1.5 References

1.5.1 Google Sign In API

Author: Google

Version Number: Not Specified

Date:

Source:

1.5.2 Google Calendar API

Author: Google

Version Number: Not Specified

Date:

Source:

1.5.3 Google Code Labs for Android Development

Author: Google

Date:

Source:

1.5.4 Android Studio Documentation

Author:

Date:

Source:

# Overall Description

The ATOMS application has been conceived as an Android application for use on Android phones at or above OS level 4.0.3 Ice Cream Sandwich. The AOMS application establishes a connection to the user’s Google account through an initial sign in screen the first time the app is launched. Once logged in, the user will be taken to a screen showing the user’s events for today’s date with a floating action button in the bottom corner. The floating action button gives the user to add auto-scheduled events to their calendar, or strict events (those events with a defined start and end time). The main screen will also have an options button that will allow the user to change the theme of the app as well as sign out, switch user accounts, or change the calendar view to a daily, weekly or monthly view. The user may also use gesture controls (swipe left, or right) to navigate throughout their calendar.

## 2.1 Product Perspective

Many applications have designed for the task of time Management, but many of them feel unintuitive and require too much effort to justify consistent use. The ATOMS development team has the perspective that the ATOMS application can be a powerful time management tool that is also easy to use.

## 2.2 Product Functions

* Login: User is able to log in through their Google Account
* Calendar Views: Presents the choice of Daily, Weekly, or Monthly Calendar View
* Adding Events: The user may add a strictly scheduled event or an automatically scheduled event
* Theme Options: The user shall have the ability to change the color theme of the app
* Gesture View Switching: The user shall have the ability to use a swipe-left or swipe-right gesture to switch to the next day, week, or month’s events depending on the current view provided
* Account Switching: The user shall have the ability to switch the user account through an option in the options menu
* Data Backup: All calendar changes will be applied to the user’s Google calendar data. If the user chooses to undo their changes, they may undo the previous added event or go back to their original calendar import.

## 4.1 Functional Requirements

TODO: Make Module Separation

|  |  |
| --- | --- |
| Requirement Number | Requirement Description |
| 1 | Login shall be handled using Google Sign in API, this is necessary for pulling Google Calendar data |
| 2 | Google Login Shall be implemented on Login Activity with a Respective Layout |
| 3 | A smooth transition to the main app screen shall be created |
| 4 | The backend code shall pull user’s Google calendar data upon entry to the main app screen |
| 5 | The main app screen center view shall consist of calendar events for the current day placed into dynamically generated and animated cards |
| 6 | The user shall be able to view the current day’s events, navigate to the next day’s event, or navigate to the previous day’s events by gesture control |
| 7 | The main app screen shall have an easily accessible button to add a new statically scheduled event or an auto scheduled event series |
| 8 | The app shall handle the users request to create a new statically scheduled event and add it into the user’s calendar |
| 9 | The app shall handle the users request to create a new auto scheduled event series and auto schedule the event, given the current calendar and event details |
| 10 | The user shall be able to enter a goal of something they would like to achieve through a quick access button on the main app screen |
| 11 | The app shall give users the ability to easily access a quick context menu to change the theme of the app or change the current view of the app (Monthly, Weekly, Daily) |
| 12 | The App shall provide a screen for users to choose between a set of predefined themes or create their own custom theme |
| 13 | The app shall not modify those existing calendar events on first input to prevent scheduling errors, unless directly authorized by the user |
| 14 | The app shall robustly handle invalid scheduling requests (scheduling two events at the same time, etc.) |
| 15 | The app shall provide OS level notifications for upcoming events, important messages, and friendly reminders |