**PROJECT SPECIFICATION**

**Schedule360**

**PURPOSE**

This document describes the project specification for the creation of Schedule360 Application. The reason for including this step in the project is to provide an accurate description of what the project plans to achieve. The project specification will aid in validating the design of the Schedule360 application.

**SCOPE**

The project specification provides an accurate description of what the project aims to achieve, its parameters and its outputs. This document works in parallel with the process specification and will be used to clearly identify the project specification in this document will represent both Non-Functional and Functional Requirements for Schedule360.

**DEFINITIONS**

* **Generalization Relationship**: A triangular shaped arrow that represents a specialized element (the child).
* **Association Relationship:** A Structural relationship that describes a set of links connecting different objects
* **UML**: Unified Modeling Language

**REVISION**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Revision Number | Revision Reason |
| Steven Ford | 04/17/2012 | 1.0 | Project Specification Creation |
|  |  |  |  |

The objective of project specification is to provide a means by which systems engineering requirements can be documented and communicated to various stakeholders involved in the development of Schedule360 Cloud Based Scheduling Application. The intent is to use UML and NFR framework - an object oriented language, to identify, create, organize and specify each of the project requirements. The classes in this document will primarily be based on two kinds of relationships – generalization relationship and Association Relationship.

**FUNCTIONAL PROJECT SPECIFICATION**

The project specification for functional requirements will identify the functional requirements for the project. The Functional Requirements for Schedule360 application are divided into the following categories:

1. Interface
   1. The application shall allow users to synchronize information between the application and all other existing electronic calendars (personal, work, etc.).
   2. The application shall allow users to interface with it on the Apple iPhone and iPad.
   3. The application shall have a search function.
   4. The application shall allow the user to sort it by location, start date, end date, start time, and end time.
   5. The application shall allow users to modify or edit the fields of an existing appointment.
   6. The application shall allow users to delete appointments
   7. The calendar shall allow registration online from [www.360schedule.com](http://www.360schedule.com).
2. Create Appointments
   1. The application shall allow the user to create appointments.
   2. The appointment shall allow the creator to specify the subject of the appointment.
   3. The appointment shall allow the creator to specify the location of the appointment.
   4. The appointment shall allow the creator to specify the end date of the appointment.
   5. The appointment shall allow the user to specify the start time of the appointment in hours and minutes.
   6. The appointment shall allow the creator to specify the end time of the appointment in hours and minutes.
   7. The appointment shall allow the creator to place notes in the appointment.
   8. The appointment shall allow the creator to set a reminder alarm.
   9. The appointment shall allow the creator the ability to invite others via email.
   10. The appointment shall allow the creator to set future reoccurrences.
   11. The appointment shall allow the creator to set their availability as Busy, Free, Tentative, and Out of the Office.
   12. The appointment shall allow the creator to attach files to the appointment.
3. View Appointments
   1. The user shall be able to select the display for all appointments as a List, Daily, Weekly, or Monthly view
4. Update Calendar
   1. The user shall be able to specify which calendar is updated.
5. ID Conflicts
   1. The application shall provide a warning if a scheduling conflict exists.
6. Update Software
   1. The software shall be updateable.

A UML diagram of the function requirements is shown below which defines how the requirements are interconnected.

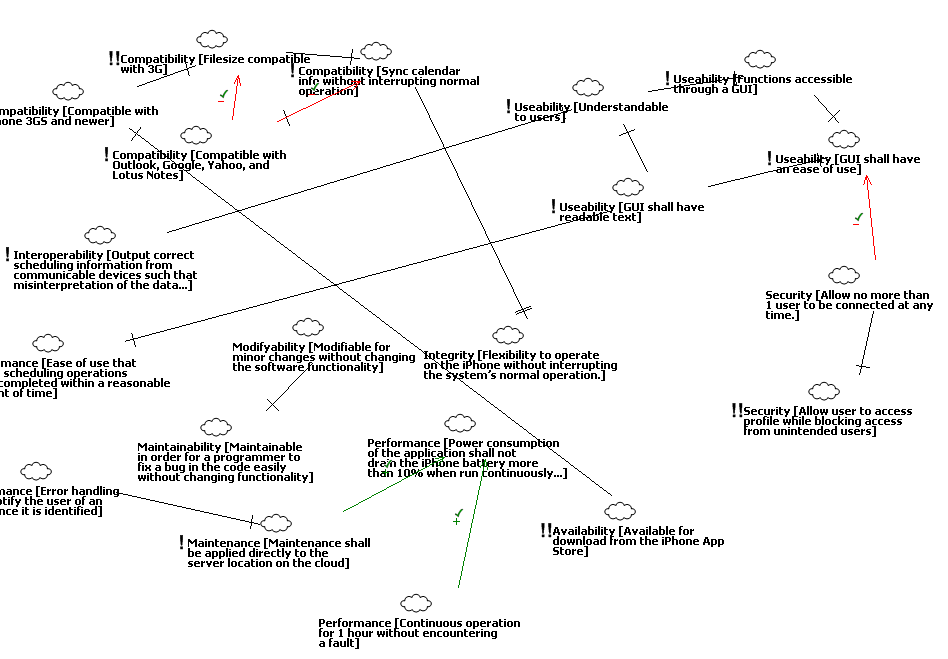


**NON FUNCTIONAL PROJECT SPECIFICATION**

The project specification for functional requirements will identify the functional requirements for the project. The Functional Requirements for Schedule360 application are divided into the following categories:

1. Compatibility
   1. The software shall be compatible as an application on the Apple iPhone version 3GS and newer.
   2. The file size of the application shall not exceed the maximum limit for iPhone apps transmitted over a 3G connection.
   3. The software shall synchronize calendar information from internet connected devices without interrupting normal operation of the connected devices.
   4. The software shall be compatible with the following calendars: Outlook, Google, Yahoo, and Lotus Notes.
2. Availability
   1. The software shall be available for download from the iPhone App Store.
3. Usability
   1. The software shall be understandable to users who perform scheduling functions on a daily basis.
   2. The software functions shall be accessible through a GUI application interface.
   3. The GUI application interface shall have an ease of use such that the user can begin performing operations without prior training or exposure to the application.
   4. The GUI application interface shall have readable text such that a user with 20/20 eyesight or corrected vision can read the text in ambient lighting from a distance of 6 inches to 24 inches.
4. Performance
   1. The power consumption of the application shall not drain the iPhone battery more than 10% when run continuously for 30 minutes.
   2. The software reliability shall allow continuous operation for 1 hour without encountering a software fault.
   3. The software error handling shall notify the user of an error once it is identified.
   4. The software shall have an ease of use that allows scheduling operations to be completed within a reasonable amount of time.
5. Integrity
   1. The software shall have the flexibility to operate on the iPhone operating system without interrupting the system’s normal operation.
6. Security
   1. The software security shall allow no more than 1 user to be connected to a personal profile at any given time.
   2. The software security shall allow a user to access his or her unique profile while blocking access to the profile from unintended users.
7. Modifiability
   1. The software code shall be modifiable in order for a programmer to make a minor change to the code easily and without changing the software functionality.
8. Maintenance
   1. The software maintenance shall be applied directly to the server location on the cloud.
9. Maintainability
   1. The software code shall be maintainable in order for a programmer to fix a bug in the code easily and without changing the software functionality.
10. Interoperability
    1. The software shall output correct scheduling information from communicable devices such that misinterpretation of the data is minimal.

A SIG diagram was created for the Non-Functional requirements using NFR framework in the StarUML tool. Diagram showing interrelationships of NFRs is shown below:

****

References

1. Supakkul, Sam. The Softgoal UML Profile: An NFRs Modeling Tool. Nov 9, 2010. Retrieved from [*http://www.utdallas.edu/~supakkul/tools/softgoal-profile/softgoal-profile.html*](http://www.utdallas.edu/~supakkul/tools/softgoal-profile/softgoal-profile.html)