Software Requirements Specification

for

Day Care Project

Version <1.0>

Prepared by

Group 1
Davis Engeler
Michael Hetzel
Jesse Leonard
John Sloan

Instructor: Dr. Schwartz

Course: Software Engineering

Date: 9/18/2014

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type and	Full Name	Information about the revision. This table does not need to be filled in whenever a document is	00/00/00

Version	Primary Author(s)	Description of Version	Date Completed
Number		touched, only when the version is being upgraded.	

<In this template you will find text bounded by the "<>" symbols. This text appears in italics and is intended to guide you through the template and provide explanations regarding the different sections in this document. There are two types of comments in this document. These comments that are in black are intended specifically for that course. These comments that are in blue are more general and apply to any SRS. Please, make sure to delete all of the comments before submitting the document.

The explanations provided below, do not cover all of the material, but merely, the general nature of the information you would usually find in SRS documents. It is based on the IEEE requirements and was adapted specifically for the needs of Software Engineering 3K04/3M04 courses. Most of the sections in this template are required sections, i.e. you must include them in your version of the document. Failure to do so will result in marks deductions. Optional sections will be explicitly marked as optional.

1 Introduction

<TO DO: Please provide a brief introduction to your project and a brief overview of what the reader will find in this section.>

1.1 Document Purpose

<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.
TO DO: Write 1-2 paragraphs describing the purpose of this document as explained above.>
This document will present a detailed description of <name> system. It will outline all of the software's features and all of its purposes, gives a clear walkthrough of each type of user interface, and the required hardware for the system. This documentation is designed to be referenced by stakeholders and the development team.

1.2 Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals.
TO DO: 1-2 paragraphs describing the scope of the product. Make sure to describe the benefits associated with the product.>

Our program will allow a transition into a user-friendly system that will help the personnel and the customers. One of the main benefits will be a consistently up-to-date view of children on site and fix organizational downfalls of the old paper system. This will be achieved through an automated sign in/out system for all children attending the day care facility. It will also improve communication between teachers and parents through classroom changes, issues that may arise, reminders from parents, illness updates and missed days, and any special need requirements for children. If any particular child is in need of timely or frequent medication it will alert the correct teacher and remind them to administer appropriately. This will increase accountability, improve and streamline record keeping for the day care, and provide an overall increase in care for the children.

Our project helps alleviate the accountability issues from a daycare center's day to day child organization routines. It offers a streamlined workflow for teachers and parents, with secure and simple sign in/out for the children.

1.3 Intended Audience and Document Overview

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers (In your case it would probably be the "client" and the professor). Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>
Development team (document writers, marketing staff, managers, and developers)
The "client" – i.e. professor
The Overall Description section describes the system in view on the

1.4 Definitions, Acronyms and Abbreviations

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.
TO DO: Please provide a list of all abbreviations and acronyms used in this document sorted in alphabetical order.>

1.5 Document Conventions

<In general this document follows the IEEE formatting requirements. Use Arial font size 11, or 12 throughout the document for text. Use italics for comments. Document text should be single spaced and maintain the 1" margins found in this template. For Section and Subsection titles please follow the template.

TO DO: Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. Sometimes, it is useful to divide this section to several sections, e.g., Formatting Conventions, Naming Conventions, etc.>

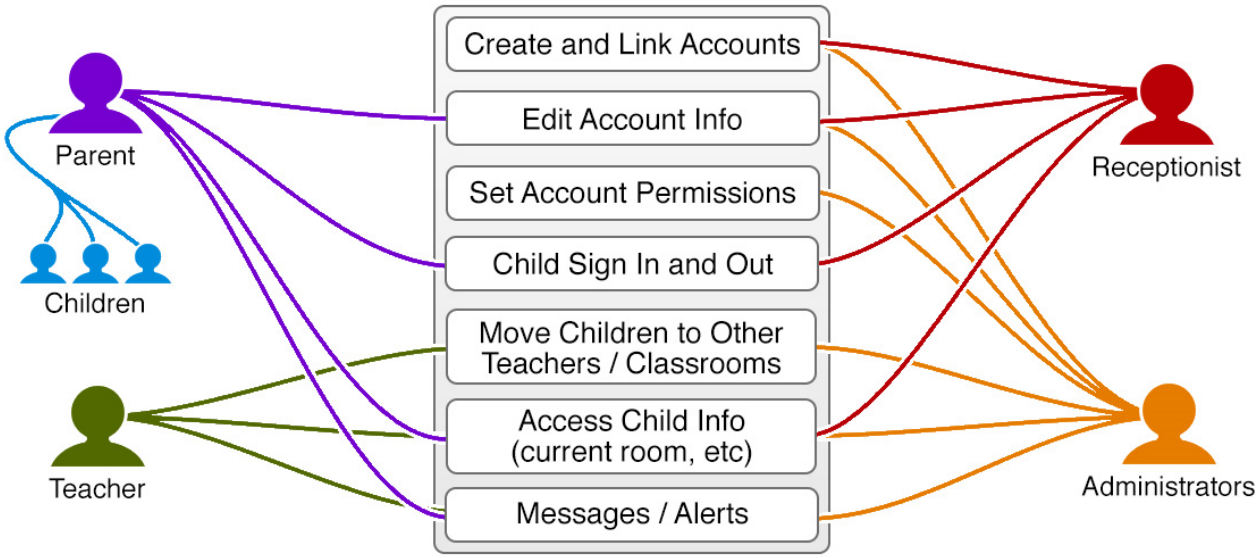
1.6 References and Acknowledgments

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document.

TO DO: Use the standard IEEE citation guide for this section. An example citation guide is posted for you on the website.>

2 Overall Description

2.1 Product Perspective



<*Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface. In this section it is crucial that you will be creative and provide as much information as possible.*

TO DO: Provide at least one paragraph describing product perspective. Provide a general diagram that will illustrate how your product interacts with the environment and in what context it is being used.>

This standalone software system originated to help alleviate the accountability issues from a daycare center's day to day child organization routines. It offers a streamlined workflow for teachers and parents, with secure and simple sign in/out for the children.

2.2 Product Functionality

<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, will be effective.
TO DO:
1. Provide a bulleted list of all the major functions of the system
2. ***(Optional)*** *Provide a Data Flow Diagram of the system to show how these functions relate to each other.>*

- Anyone will be able to sign the children in and out by using a username and password through the application. Once logged into the application there will be a sign in/out children option. The user chooses that option and selects the appropriate choice of in or out. The children's information will be automatically placed into or removed from the application and saved with a timestamp for record keeping purposes.
- The user will also have the option to use a RFID tag to sign the children in or out. It will store the account information that will allow a faster option for the checking in and out process.
- The parent client can use the Android phone app or Android tablet. The will still have the option to use the main lobby terminal.
- The added child will be assigned, in the system, to the appropriate classroom and teacher. When the parent needs to pick up the child they will have an accurate location in the day care facility.
- Inside the application, the parent can update any notes for the child. They can add anything of importance such as any illness the child may have, medications and administration the child needs, and any other special attention item. Also, they can view any notes the teacher may have left for their child and any overall class notes.
- They will receive notifications that were sent out from the day care in the form of a push notification or an email; the option will be chosen by the parent.
- Parents will receive reminders to update their personal information for annual registration. This can be accomplished in the app.
- There is an option to add or change payment methods and make payment if the parent chooses.
- The teacher, when logged in, will have a view of every child that has been signed in to the day care and is assigned to their class. This can be used for roll call whenever needed.
- If the day care needs to move a child between classes, the they will be able to select the child and choose which class to move them to. The other teacher's list will then update with the child added to their classroom list.
- If a child has been signed out of the class it will notify the teacher through a push notification. This will let the teacher know the parent is there and can respond accordingly.
- If a child has any medications to take throughout the day, it will notify the teacher at the appropriate time with how to administer the medication, the medication name, and dose.
- Also, if a parent has left any notes about the child, when the child is signed in, the teacher will get the note through a push notification so they can read it before class starts.
- Inside the app, only available to teachers and administrators, will be an option to create a message to be sent to the entire classes' parents. This will allow them to update on any events or remind parents about anything relevant to the day care class.

- The list of the students will each have an option for the teacher to send a message to just that one student's parent.

2.3 Users and Characteristics

<Identify the various users that you anticipate will use this product. Users may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience.
TO DO:
1. Describe the pertinent characteristics of each user. Certain requirements may pertain only to certain users.
3. Distinguish the most important users for this product from those who are less important to satisfy.>

Main Lobby Access:

- Allows anyone to sign the children in and out by using a username and password through the application. Once logged into the application there will be a sign in/out children option. The user chooses that option and selects the appropriate choice of in or out. The children's information will be automatically placed into or removed from the application and saved with a timestamp for record keeping purposes.
- The user will also have the option to use a RFID tag to sign the children in or out. At the Main Lobby Terminal the tag will be read and it will automatically add or remove the children based on the current status of the child without the need to actually sign in to the account or choose any option.
- Another option on the Main Lobby Terminal after the parent signs in will be to view the child's current classroom location. If the parent needs to pick up the child they will have an accurate location in the day care facility.
- Inside the application, the parent can update any notes for the child. They can add anything of importance such as any illness the child may have, medications and administration the child needs, and any other special attention item.

Parent Client

- The parent client can use the phone app on their own personal phone versus using the main lobby terminal.
- They will receive notifications that were sent out from the day care and can see the current room their child/children are in. If a child switches classroom they will be able to see the change.
- They can add notes for the day care and teacher about anything that is relevant to their child while they are attending. Also, they can view any notes the teacher may have left for their child and any overall class notes.
- Parents will receive reminders to update their personal information for annual registration. This can be accomplished in the app.
- There is an option to add or change payment methods.

Teacher Client

- The application will be able to run on Android tables or cellular phones.
- The teacher, when logged in, will have a view of every child that has been signed in to the day care and is assigned to their class. This can be used for roll call whenever needed.
- If the day care needs to move a child between classes, the teacher will be able to select the child and choose which class to move them to. The other teacher's list will then update with the child added to their classroom list.

- If a child has been signed out of the class it will notify the teacher through a push notification. This will let the teacher know the parent is there and can respond accordingly.
- If a child has any medications to take throughout the day, it will notify the teacher at the appropriate time with how to administer the medication, the medication name, and dose.
- Also, if a parent has left any notes about the child, when the child is signed in, the teacher will get the note through a push notification so they can read it before class starts.
- Inside the app, only available to teachers and administrators, will be an option to create a message to be sent to the entire classes' parents. This will allow them to update on any events or remind parents about anything relevant to the day care class.
- The list of the students, the teacher has access to, will each have an option for the teacher to send a message to just that one student's parent.

Receptionist/Staff Client

- Responsible for adding parents and children to the system with the needed information. Will be approved by administrative client.
- If there needs to be a change to the children's or parent's information, they can change it for them.
- If there is a problem with checking a child in or out, the staff can do it for the parent.
- At any point in time, they can see where the child is currently and any information the child might have.

Administrative Client

- Ability to add, edit, remove, or view all content within the application.
- Any of the options for functions that any user can execute, the administrative client will be able to use as well.
- Responsible for reviewing new staff, teacher, and parent user accounts and changing account rights and permissions accordingly.

2.4 Operating Environment

<*Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface*
TO DO: As stated above, in at least one paragraph, describe the environment your system will have to operate in. Make sure to include the minimum platform requirements for your system. >

2.5 Design and Implementation Constraints

<*Describe any items or issues that will limit the options available to the developers. These might include: hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).*
TO DO: In this section you need to consider all of the information you gathered so far, analyze it and correctly identify at least 5 constraints.>

2.6 User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.
TO DO: You will not actually develop any user-manuals, but you need to describe what kind of manuals and what kind of help is needed for the software you will be developing. One paragraph should be sufficient for this section.>

2.7 Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project.
TO DO: Provide a short list of some major assumptions that might significantly affect your design. For example, you can assume that your client will have 1, 2 or at most 50 Automated Banking Machines. Every number has a significant effect on the design of your system. >

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., Cancel) that will appear on every screen, error message display standards, and so on. Define the software components for which a user interface is needed.
TO DO: The least you can do for this section is to describe in words the different User Interfaces and the different screens that will be available to the user. Those who will be able to provide optional Graphical User Interface screenshots, will be rewarded by extra marks.>

3.1.2 Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware. You are not required to specify what protocols you will be using to communicate with the hardware, but it will be usually included in this part as well.
TO DO: Please provide a short description of the different hardware interfaces. If you will be using some special libraries to communicate with your software mention them here. In case you have more than one hardware interface divide this section into subsections.>

3.1.3 Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems (Windows? Linux? Etc…), tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.

TO DO: The previous part illustrates some of the information you would usually include in this part of the SRS document. To make things simpler, you are only required to describe the specific interface with the operating system.>

3.1.4 Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.

TO DO: Do not go into too much detail, but provide 1-2 paragraphs were you will outline the major communication standards. For example, if you decide to use encryption there is no need to specify the exact encryption standards, but rather, specify the fact that the data will be encrypted and name what standards you consider using. >

3.2 Functional Requirements

< Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. This section is the direct continuation of section 2.2 where you have specified the general functional requirements.

Here, you should list in detail the different product functions with specific explanations regarding every function.

TO DO: Brake the functional requirements to several functional areas and divide this section into subsections accordingly. Provide a detailed list of all product operations related to these functional areas.

3.3 Behaviour Requirements

3.3.1 Use Case View

<A use case defines a goal-oriented set of interactions between external actors and the system under consideration. Since sometimes we will not be able to specify completely the behaviour of the system by just State Diagrams, we use use-cases to complete what we have already started in section 3.3.1.

TO DO: Provide a use case diagram which will encapsulate the entire system and all possible actors. Do not include detailed use case descriptions (these will be needed when you will be working on the Test Plan), but make sure to include a short description of what every use-case is, who are the actors in your diagram. For more information please refer to your UML guide and the MiniThermostat SRS example file.>

4 Other Non-functional Requirements

4.1 Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.
TODO: Provide at least 5 different performance requirements based on the information you collected from the client. For example you can say "1. Any transaction will not take more than 10 seconds, etc…>

4.2 Safety and Security Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product's design or use. Define any safety certifications that must be satisfied. Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements.
TODO:

- *Provide at least 3 different safety requirements based on your interview with the client or, on your ABM related research, and again you need to be creative here.*
- *Describe briefly what level of security is expected from this product by your client and provide a bulleted (or numbered) list of the major security requirements.>*

4.3 Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.

TODO: Use subsections (e.g., 4.3.1 Reliability, 4.3.2 Portability, etc…) provide requirements related to the different software quality attributes. Base the information you include in these subsections on the material you have learned in the class. Make sure, that you do not just write "This software shall be maintainable…" Indicate how you plan to achieve it, & etc…Do not forget to include such attributes as the design for change. Please note that you need to include at least 2 quality attributes, but it is the mere minimum and it will not receive the full marks.>

5 Other Requirements

*<This section is **Optional**. Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

Appendix A – Data Dictionary

<Data dictionary is used to track all the different variables, states and functional requirements that you described in your document. Make sure to include the complete list of all constants, state variables (and their possible states), inputs and outputs in a table. In the table, include the description of these items as well as all related operations and requirements.>

Appendix B - Group Log

<Please include here all the minutes from your group meetings, your group activities, and any other relevant information that will assist the Teaching Assistant to determine the effort put forth to produce this document>