Software Requirements Specification

for

Scripture Typing

Version 2.0

Prepared by

Group Name: To Be Decided

Jennifer Weeks	<student #>	JenniferWeeks@letu.edu
Reed Lawrence	<student #>	ReedLawrence@letu.edu
Joshua Wilfong	<student #>	JoshWilfong@letu.edu

Instructor: Dr. Brent Baas

Course: Software Engineering

Date: 2/18/2015

Contents

REVISIONS ..... **III**

1 INTRODUCTION **1**

1.1 DOCUMENT PURPOSE 1
1.2 PRODUCT SCOPE 1
1.3 INTENDED AUDIENCE AND DOCUMENT OVERVIEW 1

2 OVERALL DESCRIPTION **2**

2.1 PRODUCT FUNCTIONALITY 2
2.2 OPERATING ENVIRONMENT 2
2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS 3
2.4 USER DOCUMENTATION 3

3 SPECIFIC REQUIREMENTS **4**

3.1 EXTERNAL INTERFACE REQUIREMENTS 4
3.2 FUNCTIONAL REQUIREMENTS 4

APPENDIX A – DATA DICTIONARY **5**

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
First Draft	TC Buscher Reed Lawrence Caleb Phillips	Primary Draft	2/4/2015
2.0	Jennifer Weeks Reed Lawrence Joshua Wilfong	Adding functional requirements and diagrams	2/18/2015

1 Introduction

1.1 Document Purpose

Document encompasses the entire project given by the customer.

1.2 Product Scope

Product will help improve typing speed and accuracy. Users will practice typing Bible verses and through repeated typing will become more familiar and eventually memorize the verses.

1.3 Intended Audience and Document Overview

The primary audience is 16 and above (i.e. those old enough to deal with specifics about different spiritual issues), though less topic-specific “kids' game” versions may be designed for children 8–16 (i.e. those old enough to type and comprehend what they are typing).

2 Overall Description

2.1 Product Functionality

- Small database of verses for each course
    - Random verses generated for each encounter
    - Verses are "tagged" for different topics

Player will be able to select in the start menu the different tags, from 1 to all being available for selection, before pressing start. During the game only verses with those tags will generate for their encounters. Each verse may have multiple tags. Each single tag must be supplied with enough verses to complete the journey.

- Main Menu
    - Selectable font sizes
        - For those who have trouble reading smaller fonts
    - Typing Speed Goal
        - This will dictate the difficulty of the encounter. The higher the goal the faster your enemies will move during the encounter.
        - Optional: adjustment of typing speed goal during game in case user goal is unrealistic for their abilities
    - Length of Journey
        - See game length below
    - Verse Topics
        - Selected as check boxes, players will have the option of selecting as many as they want, but must select at least one.
        - A select all button will be available.
    - Start
- Turn sequence
    - Hit start
    - Static background (leading to hill with cross)
    - Character automatically walks along path until encounter
    - Encounter
        - Find and opponent and a verse appears at top or bottom of screen
        - Begin typing verse as quickly as possible
        - Opponent advances at slow set pace
        - Correct character input pushes opponent back
        - Incorrect character input allows opponent to advance
        - Branch: victory or defeat
    - Branch victory – resume automatic walk
    - Branch defeat – slowed/injured
        - Injury is removed by redoing battle (no penalty for loss here)
- Game Length

    - Demo: 2 encounters (Roman Road)
    - Quick Game: 3-5 encounters
    - Regular Game: 10-12 encounters
    - Long Game: 24-26 encounters
- Typing stats
    - Speed
    - Accuracy

2.2 Operating Environment

Desktop running Windows 7 (possibly working on Mac OS X as well). No mobile versions planned. Optional web version compatible with all operating systems.

2.3 Design and Implementation Constraints

ESV Bible Translation
Optional: access online database for other versions of the Bible.

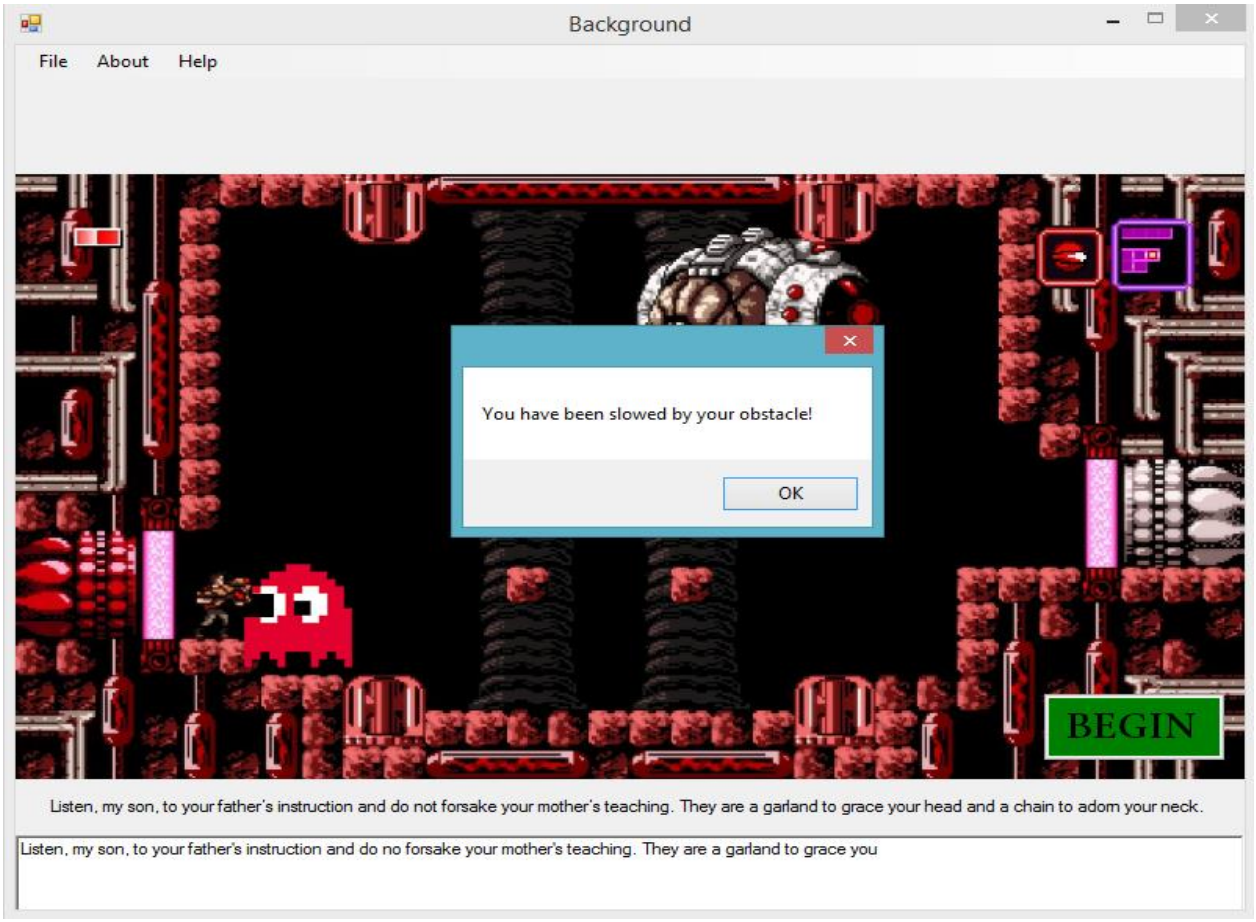
2.4 User Documentation

Game instructions/help page.

3 Specific Requirements

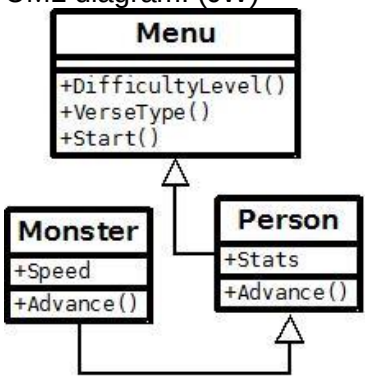
3.1 External Interface Requirements

3.1.1 User Interfaces

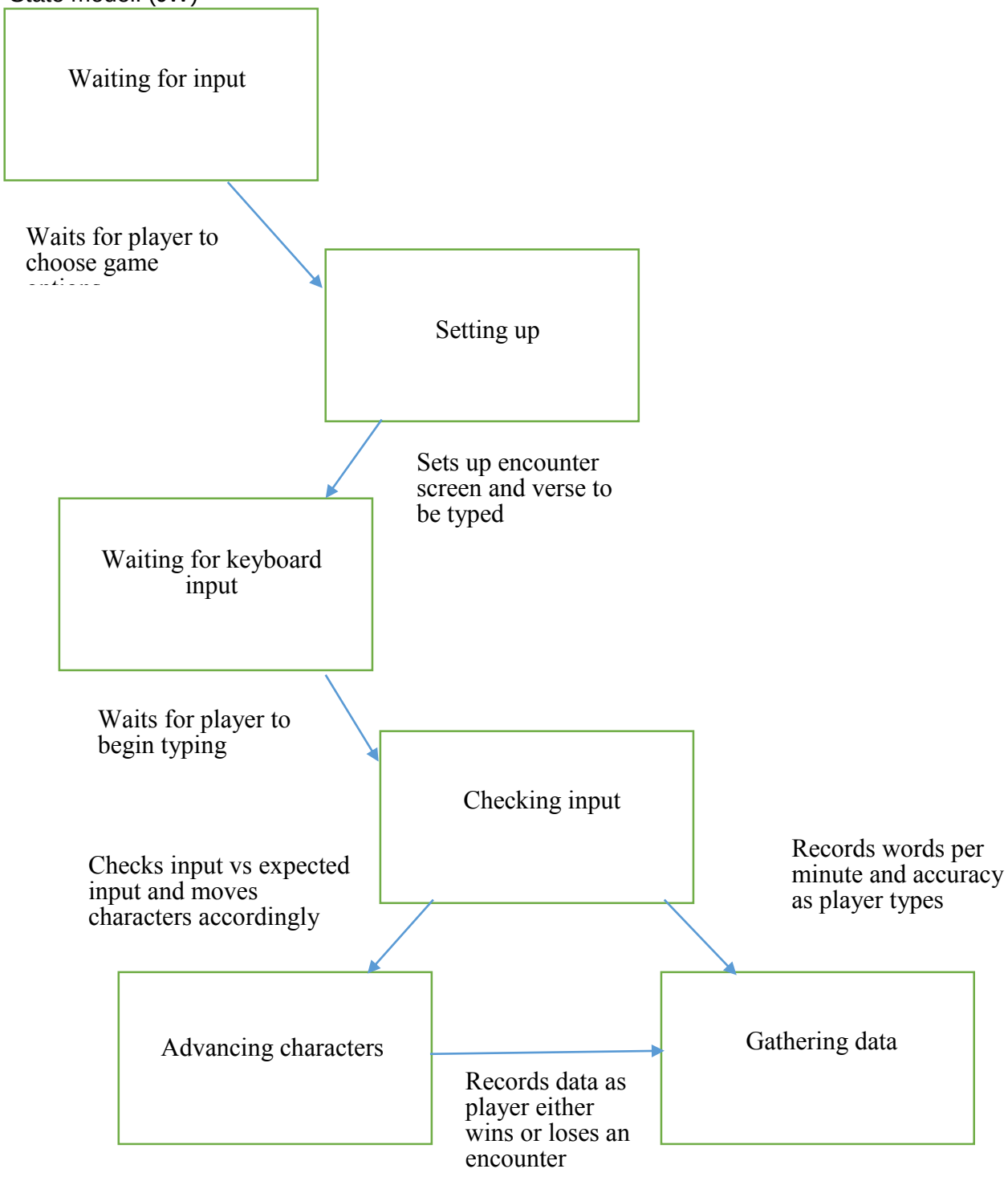
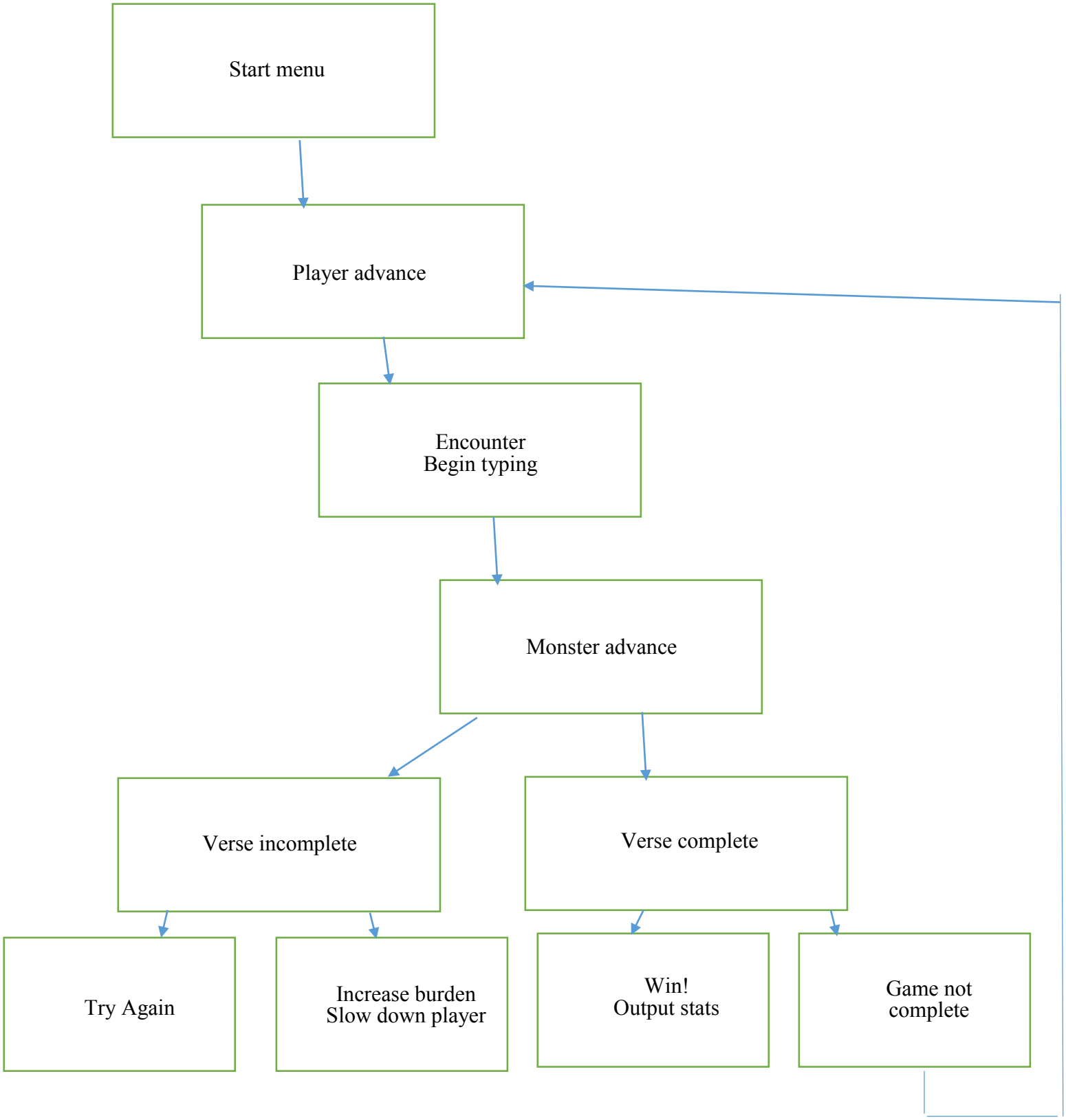


3.2 Functional Requirements

UML diagram: (JW)



State model: (JW)



Use Case model: (JW)