

Ninja Typers Functional Specification

Functional Specification

By

Marvin Farrell

Table of Contents

1. Objectives	2
1.1. Project Objective	2
1.2. Account Management System	2
1.3. Typing Web Application	2
1.4. Database	2
1.5. User Interfaces	3
1.6. Cloud	3
2. Functionality	3
2.1. Functionality Description	3
2.2. External Interfaces	4
3. Application Users and Usage	4
3.1. Site Administrator	4
3.2. Teacher/Administrator	4
3.3. Student/Child	5
3.4. Use Case Diagram	5
3.5. Use Cases	6
3.6. Detailed Use Cases	7
4. Similar Applications	9
4.1. TypingClub	9
4.2. How does this application differ?	10
4.2.1. Paid	10
4.2.2. Administrators	10
4.2.3. Grading Structure	10
5. Supplementary Specification	10
5.1. Functionality	10
5.2. Usability	11
5.3. Reliability	11
5.4. Performance	11
5.5. Supportability	11
6. References	12

1. Objectives

1.1. Project Objective

The objective of the project is to provide a child with an environment to teach them how to touch type or to build on their typing skillset. The project also has an account management system in to manage accounts of teachers and students using the system.

1.2. Account Management System

The objective of the account management system is to provide a system with the functionalities needed to manage accounts of people who wish to use the system.

For a student using the system the objective is to provide them with the ability to do the following:

- View account.

For a teacher the objective is to provide a system that will allow the teacher to do the following:

- Create account.
- Create, update or delete a student account.
- Monitor student accounts.
- Make payment for system use.

For a site administrator the objective is to provide a means of activating any accounts that have registered for use of the system.

1.3. Typing Web Application

The objective of the typing web application is give the student an environment to learn at their own pace in an environment that is suited to their needs and ability.

Another objective of the web application is to track all statistical information relating to the current lesson being taking i.e. word count, number of correct/incorrect keystrokes, time etc..

1.4. Database

The object of the database is to provide the storage facility needed for all data relating to users of the system and for information on the course content also.

1.5. User Interfaces

There are a number of objectives for the user interfaces for this system. The first is to provide the user with a look and feel that will drastically decrease the complexity of using the system as there will be children using the system.

The second is to provide a simple way to navigate through different areas of the system. This is more important for the younger people using the system.

1.6. Cloud

The objective of the cloud is to provide a place to develop the application and environment to host the application. This means providing a domain to run the web application and storage location for the database.

2. Functionality

2.1. Functionality Description

- ***RegisterAccount***
For a new user to the system the RegisterAccount function will control the creation of a new administrative account. A new instance of a teacher will be created and added to the database where it waits to be activated.
- ***ActivateAccounts***
The site administrator uses this function to retrieve information from the database and activate the user account of somebody who has registered a new account to use the system.
- ***AddStudent***
The AddStudent function will be used by a teacher/administrator to add a new student to his/her course. It will create a new student instance and add it to the database on the cloud.
- ***UpdateAccount***
When a teacher wishes to manually update the accounts of students enrolled on their course the UpdateAccount function will be used to update their information stored on the cloud.
- ***DeleteAccount***
If a situation arises and the account of a student enrolled on a course needs to be removed this function is used to erase that data from the database.

- **Login**
This function controls the login part of the system by checking the information on the page with information stored on the database for a user account.
- **Logout**
This function controls the logout part of the system.
- **ViewAccount**
The function will retrieve account information for the user who is logged in and will populate the web page with that information.
- **SelectLesson**
The SelectLesson function is used to select the lesson from the database to be used in the typing lesson.
- **Make Payment**
This function is used to handle payments for using the system.

2.2. External Interfaces

- **User Input**
User input for the system will vary depending on the type of user.
 - Student: User input
 - Input for logging in to the system.
 - Input for typing activity.
 - Teacher: User input
 - Input for logging in to the system.
 - Input for creating a new account.
 - Input for adding a new student account.
 - Input for updating account.
 - Input for making payments for system use.
 - Site Administrator: User Account
 - Input for logging into the system.
 - Input for creating a new lesson.

- o Input for activating accounts.

3. Application Users and Usage

3.1. Site Administrator

The site administrator will use the application for a number of purposes.

- o Account activation: The site administration will use the system to activate the accounts of people who have subscribed to use the system.
- o Create lesson: The site administrator will use the system to create new lessons for students to use in the typing activity.

3.2. Teacher/Administrator

The teacher will use the application for account management.

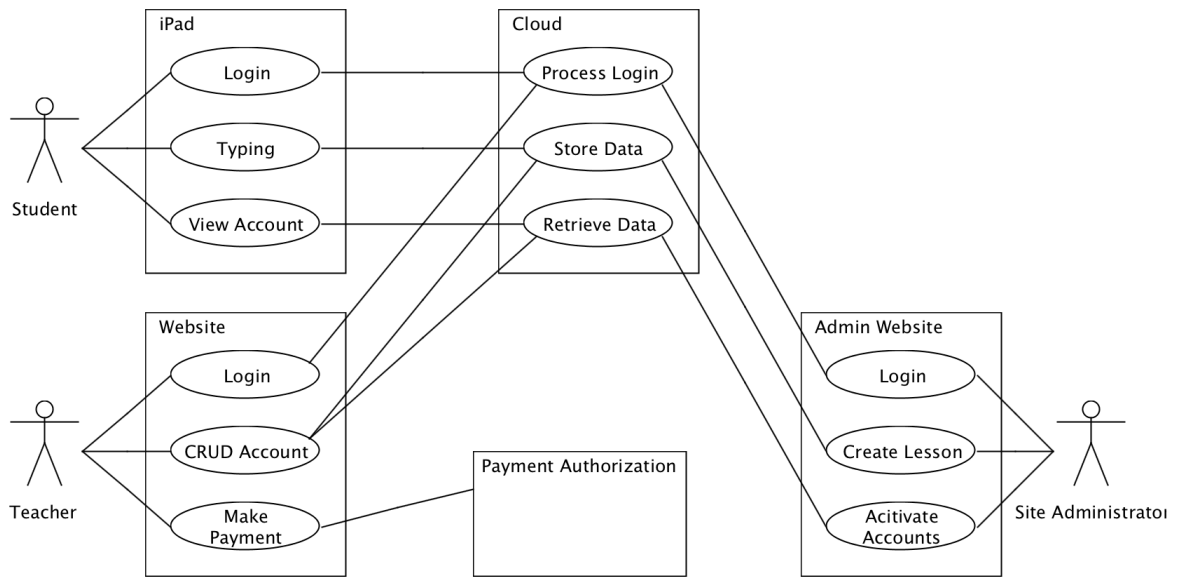
- o Account registration: The teacher will use system to register new accounts.
- o Add students: The teacher will use the system to create student account and add them to their course.
- o Delete accounts: The teacher will use the system to remove student account from the course.
- o Update account: The teacher will use the system to manually update student account information.
- o Payment: The teacher will use the system to pay for student/students use of the system.
- o Student account monitoring: The teacher will use the system to monitor the accounts of student.

3.3. Student/Child

The student will use the system to two main purposes.

- o View account information: The student will use the system to view information on their account to see course progression.
- o Typing: The student will use the web application for typing activities.

3.4. Use Case Diagram



3.5. Use Cases

Login

Actors: Student, Teacher, Site Administrator, Cloud.

This use case begins when the student, teacher or site administrator attempts to login. The student, teacher or site administrator enters their details into the login page and if the login is successful are redirected to the home page.

Typing

Actors: Student, Cloud.

This use case begins when the student starts a typing lesson. The student has to type all the keystrokes that are in the current lesson. When the typing lesson is complete, stats from the current lesson is uploaded to the cloud.

View Account

Actors: Student, Cloud.

This use case begins when a student wishes to view information regarding their own account such as name, score, number of words types, current certificate, accuracy etc. The student's data is retrieved from the cloud and displayed on the page.

CRUD Account

Actors: Teacher, Cloud.

This use case begins when the teacher attempts to create, read, update or delete a student account. When the teacher carries out the selected option the database on the cloud is updated.

Make Payment

Actors: Teacher, Payment Authorization.

This use case begins when the teacher attempts to make a payment for the student\students using the application. The teacher enters the payment details and this is sent to the Payment Authorization to make the payment.

Activate Accounts

Actors: Site Administrator, Cloud.

This use case begins when the site administrator attempts to activate any un-activated teacher or student account stored on the cloud.

Create Lesson

Actors: Site Administrator, Cloud.

This use case begins when the site administrator attempts to create a new lesson for a specific grade. When the lesson is created it is stored on the cloud.

Process Login

Actors: Student, Teacher, Site Administrator, Cloud.

This use case begins when information is sent from the student, teacher or site administrator for a login. The cloud checks this information with data stored in the database on the cloud.

Retrieve Data

Actors: Student, Teacher, Site Administrator, Cloud.

This use case begins when information is requested from the student, teacher or site administrator for an activity. The data is retrieved from the database and sent back to the student, teacher or site administrator.

Store Data

Actors: Students, Teacher, Site Administrator, Cloud.

This use case begins when information is sent from the student, teacher or site administrator to the cloud. The data is stored in the database on the cloud.

3.6.Detailed Use Cases

Login:

Actors: Student, Teacher, Site Administrator, Cloud.

Main Success Scenario:

1. Student, Teacher Site Administrator enters credentials into the login area.
2. Student, Teacher or Site Administrator press enter button.
3. Cloud processes login.
4. If login is successful Student, Teacher or Site Administrator is redirected to their home page.

Typing:

Actors: Student, Cloud

Main Success Scenario:

1. Student selects the typing lesson they want to take.
2. Student selects the option to start typing.
3. Student types all keystrokes for the current lesson.
4. Upon completion the information for lesson is saved.
5. Information is sent to the cloud.
6. The cloud stores the data.
7. Student account is updated.
8. The Student can pick the option to return to their home page, retake the lesson or proceed to the next lesson.

View Account:

Actors: Student, Cloud

Main Success Scenario:

1. Student selects the option to view their account information.
2. The Cloud retrieves data.
3. Information is displayed on the Student page.

CRUD Account:

Actors: Cloud, Teacher

Main Success Scenario:

1. Teacher goes to Student accounts.
2. Cloud retrieves data.
3. All students on course accounts are displayed.
4. If the Teacher selects the option to create account account.
 - a. Teacher enters student information in new accounts fields.
 - b. Teacher presses create button.
 - c. Information is sent to the cloud.
 - d. Cloud stores the information.
 - e. Account awaits activation.
5. If the Teacher selects the option to update account.
 - a. Teacher selects the account to update.
 - b. Teacher modifies the information of the account.
 - c. Teacher presses the update button.
 - d. Information is sent to the cloud.
 - e. Cloud stores the information.
6. If the Teacher selects the option to delete account.
 - a. Teacher selects the account to delete.
 - b. Teacher presses the delete button.
 - c. Information is sent to the cloud.
 - d. Cloud stores the information.

Create Lesson:

Actors: Site Administrator, Cloud.

Main Success Scenario:

1. Site administrators select the option to create new lesson.
2. Site administrator selects the grading certificate of the lesson.
3. Site administrator types in all the characters to be used in the lesson.
4. Save button is pressed.
5. Lesson is sent to the cloud.
6. Cloud stores information.

Activate Accounts:

Actors: Site Administrator, Cloud.

Main Success Scenario:

1. Site Administrator selects the accounts option on the home page.
2. Data is retrieved from Cloud.
3. Accounts are displayed on page.
4. Site Administrator selects the account.
5. Site administrator presses the activate account button.
6. Data is sent to the Cloud.

7. Cloud stores data.

Process Login:

Actors: Cloud, Student, Teacher, Site Administrator

Main Success Scenario:

1. Cloud receives login information from Student, Teacher or Site Administrator.
2. Cloud checks data with data stored on database for a match.

Store Data:

Actors: Cloud, Student, Teacher, Site Administrator

Main Success Scenario:

1. Cloud receives data from Cloud, Student, Teacher or Site Administrator.
2. The account for the Cloud, Student, Teacher or Site Administrator is located.
3. The account is updated with the received data.

Retrieve Data:

Actors: Cloud, Student, Teacher, Site Administrator

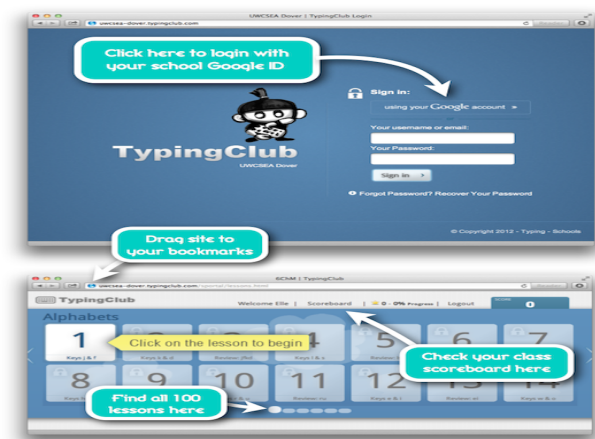
Main Success Scenario:

1. Cloud receives request for data from Student, Teacher or Site Administrator.
2. Cloud locates the account for the Student, Teacher or Site Administrator.
3. The requested data is retrieved from that account.

4. Similar Applications

4.1. TypingClub

TypingClub is an online web application that has a lot of similarities to the application that will be developed. The only language that is used in TypingClub is English. The application has a free option but also has a full version for educational purposes called TypingClub School edition which does have a fee associated with it which works out at about \$2.99 per student using the application. The application start the user off by using two keys initially but as the user progresses through the level of difficulty increases by incorporating more keys and alternative fingers into the typing activity. Using the free edition the user can track their own progress as the advance through the lessons. Using the school edition the progress can be monitored by an administrator like the teacher. The free edition has the lessons fixed so everybody will get the same content but the school edition has the option to create custom content to use in the classroom. There is no need to create an account for use of the free edition but it is advised to create an account to keep track of data collected from using the application.



Typing activity screenshot

Account login/Level selection screenshot

4.2. How does this application differ?

4.2.1. Paid

Unlike TypingClub this application has a fee associated with usage. There is no free option for use of the application. The main reason that this application has a charge associated with it is that it is going to be used in NinjaTypers which provides members with the hardware needed participate in the course.

4.2.2. Administrators

Because this application is being developed for children to touch type it is crucial that their progress and activity is being monitored by an administrator like a parent or teacher. Other applications present the user with the option to create an account and start using the application. This application takes a different approach. An administrator account must first be set up and then a child using the application at home or student account can be

associated with the administrative account. Then the person can use the application for typing activities.

4.2.3. Grading Structure

The grading structure that will be used in the application is one that has been designed and is currently being used at NinjaTypers. This is a structure that has proven to be very effective with people taking the course with little or no typing skills building up a typing skill set and progressing through the course at a very fast consistent pace.

5. Supplementary Specification

5.1. Functionality

All functionality of the entire system should be available through an internet connection. All system errors should be recorded and logged.

5.2. Usability

The Site Administrator (Administrator) should be able to build up the skillset to create documents, deliver the lesson and check statistics of the children within an hour of using the site. The child should be able to navigate their way through the different pages within 5 minutes and should be competent in using all features on the site after about 15 minutes. The application should run on all web browsers.

5.3. Reliability

Should never be an issue with access to the site. The site should be available 24 hours per day, 7 days a week with a maximum downtime of 6%.

5.4. Performance

Response time of the application should not be more than 1 second. There should be easy transitions between pages. Performance speed of the application will depend on the speed of the internet connection. Database access response time should not have latency greater than 10 seconds, anything higher is not acceptable.

5.5. Supportability

Testing for the application will be on-going through the implementation stages of the project. The application is geared for desktop/laptop for the person who is delivering the content and delivered to a tablet device with iPads currently being used for children in the club.

6. References