**DOT Design Document (DDD)**

Written by:

Hieu Tran

Latest Revision

December 2, 2017

# Product Overview & Summary

The fact of the matter is that typing skill is an important thing to learn, at any age, to make you a more coveted employee. DOT’s purpose is to increase the user’s WPM and accuracy by building muscle memory.

# Component Overview & Summary

This section provides a general description of this document’s purpose and the material that the document is going to cover. It may also provide a general overview of the component, its features, and its intended users.

* User Profile Database
* Testing Hub
* Typing Tests

**User Profile Database:**

Relationships:

* Testing Hub feeds info from the typing tests.

**Testing Hub:**

Relationships:

* Sends data to the User Profile Database

**Typing Tests:**

Relationships:

* Takes integers generated from typing tests

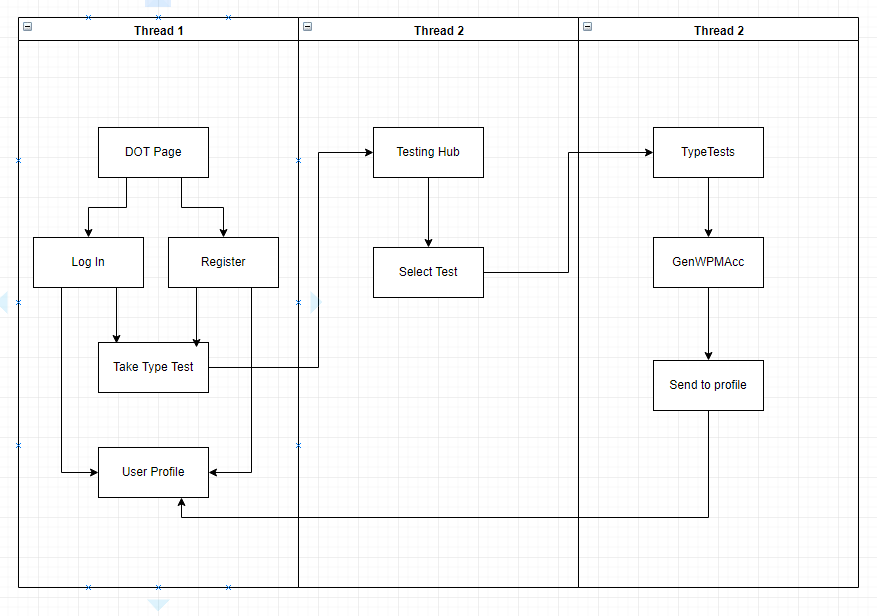
# Reusable Components Document

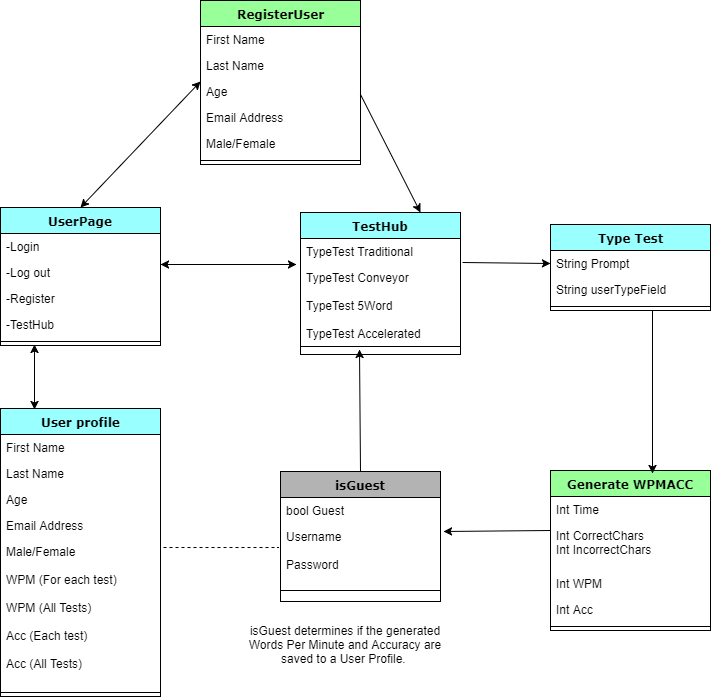
* User profile
  + Fed info from the typing tests
  + Displays WPM and Accuracy for each/all typing tests.
  + Must be created from the register page on the Hub.
* Type Tests
  + Generates/pulls words or passages from database depending on type of tests the user takes.
  + Calculates WPM and Accuracy of the user in real time during the typing tests.
* Database
  + Contains passages used to fill word bank for Type Tests.

# Requirements

### Data Flow

This section shows a “high-level” data flow diagram of the component.





## User-Case Requirements

The section contains the bulk of the material in the document. Use-Case requirements outline the details of what occur.

* Guest is not signed in and takes a traditional test. (Finishes it)
  + They will start from the hub page and select tests.
  + User is given a list of different tests they can take.
    - Different use cases for each test where the only difference for each test is the specific test itself and where the information about the test is saved.
  + The user types the entire passage correctly.
  + Nothing is saved because they are not logged in.
* Guest is not signed in and takes a traditional test. (Does not finish it)
  + They will start from the hub page and select tests.
  + User is given a list of different tests they can take.
  + The user does not finish the type test.
  + Nothing is saved because they are not logged in *as well* as never having finished the test..
* User wants to sign up for an account.
  + The user lands on the hub interface.
  + User clicks on the sign up button.
  + They enter all of their information and a new user profile is created
    - Else, the sign up page will tell the user that they are missing something.
* Becky logs in, takes a test but does not finish it.
  + She lands on the hub interface.
  + User clicks the sign in button
  + Dropdown block for username and password from the sign in button.
  + After successful login, she will be back on the hub but now signed in.
    - In the event of incorrect login, user will get a message about an incorrect login.
  + She takes a typing test but stops before the timer ends.
  + Test is terminated and data is not saved.
* Becky logs in, takes a test and finishes it
  + She lands on the hub interface.
  + User clicks the sign in button
  + Dropdown block for username and password from the sign in button.
  + After successful login, she will be back on the hub but now signed in.
    - In the event of incorrect login, user will get a message about an incorrect login.
  + She takes a typing test and inputs all correct keystrokes.
  + WPM and accuracy is calculated and saved into her user profile.
* Guest makes an incorrect keystroke and keeps typing.
  + User decides to take a traditional typing test
  + User makes incorrect keystroke on a character
  + System allows him to keep typing in the field but does not advance the string conveyor belt.
  + User types all available characters despite having wrong keystroke
  + Test is terminated because user is not logged in.
* Sam makes an incorrect keystroke and keeps typing.
  + She lands on the hub interface.
  + User clicks the sign in button
  + Dropdown block for username and password from the sign in button.
  + After successful login, she will be back on the hub but now signed in.
    - In the event of incorrect login, user will get a message about an incorrect login.
  + User decides to take a traditional typing test
  + User makes incorrect keystroke on a character
  + System allows him to keep typing in the field but does not advance the string conveyor belt.
  + User types all available characters despite having wrong keystroke
  + Test is saved. WPM and accuracy are calculated and saved into user profile.

## Performance Requirements

DOT will be able to handle calculations of WPM and Accuracy in real time. The only lag that comes from the interface experience will be ones that involve transferring of data between database, server, and user.

## Foreseeable Modifications & Enhancements

DOT may or may not be an ongoing process. Future modifications are unclear.