**Product Requirements Document**

1. **Introduction: The Product**

Branding Pays is Karen Kang’s methodology and toolset used in her workshops with companies on personal branding. We will be building the web application that comprises the toolset. This application will:

* Provide supplemental interactive templates that workshop members can fill out online
  + These templates will provide a place for members to create their personal brand in writing
* Store user data on a server
* Be secure
  + It will only allow access to members of the workshops and their data will be kept private and secure from hackers

2. **User Interface**

2.1 Brand Considerations

The UI will reflect the standards outlined in Karen Kang’s “BrandingPays Brand, Tools, and Site Considerations” document. Some include:

* Look and Feel: Complementary to but different from Karen Kang Consulting (website)
* Colors: Same as color palette from KKC website and PowerPoint presentations
* Personality:
  + Classy, professional and high quality
  + Personable, approachable and engaging
  + Unique and attractive with attention to detail
  + Innovative and Fresh
  + In keeping with Karen Kang’s values of providing high value in a quick, interactive and easy-to-understand way

2.2 UI Requirements

Navigation:

* Users must be able to navigate to any template or page they would like at any time
  + This will be accomplished by a top tab navigation bar
* Users need a way to go to the next or previous page
  + Accomplished by buttons at the bottom of the page
* The UI will be structured as “wizard,” guiding the user through the process, letting them know where they are
* There needs to be a way for users to leave the program and save
* Mouse overs as tool tips
  + Must design this with aethetics and user experience in mind

3. **Programming**

3.1 Language Requirements

* Use HTML(5)/CSS(3), JavaScript, PHP, and Dojo library to accomplish tasks
  + No Flash
* JavaScript will be needed for programming interactivity of application

3.2 Data Storage

* User data will be stored in JSON files on the server
  + When login occurs, user data will be pulled down from the server
* Data must be auto saved every n seconds (or due to certain events)

3.3 Difficult programming tasks, such as the programming of tables, will be aided by the use of Dojo

4. **Security**

* User data and passwords must be kept private and safe from hackers
  + Use token in PHP
  + Prevent Cross-Site Scripting
    - Write code that prohibits certain characters in forms, such as “<”, “>”

5. **Legal**

* BrandingPays tools must be kept safe from replication by others
* Need to instate proper copyrights and trademarks, user license agreements and terms of conditions of use

6. **Quality Control**

* Extensive user testing and feedback required
  + Part of user testing will involve sitting down with users and observing the way they interact with the application

7. **Administrative Needs**

7.1 Figure out how to grant and limit user licenses

* Limiting licenses:
  + Will licenses time out?
  + What would a time out mean?
    - Application becomes read-only
  + How long would editing license last?

7.2 Administrative Tools

* Ability for Karen to do a simple programming task to grant and limit licenses for certain companies/individuals
* Karen should be able to update the content of the application as easily as possible, on her own (maybe not possible though)
  + Alternatively, Allie could be on call to make changes
* Determine whatever other administrative tools are needed

7.3 Agreement with Programming Team

* Confidentiality and ownership of code work
* Signed by team members

8. **Login Issues**

* Should email verification be required for someone to register?
* Fields included for registration:
  + First and last name, company email, username, and password
  + Any others?
* Forgotten password/username system
  + Send email
* Guest access
  + Timed login with guest username and password
  + Two levels of guest login:
    - Able to print/save as pdf
    - Unable to print/save as pdf

9. **Bugs**

* Need bug reporting system
  + A link in the application that takes user to bug reporting page