# LunchMoneyMatch.com

# **Color Scheme:**

Undecided. We would like to keep the norm of social media colors, so keep it close to Facebook (#3b5998). The color of the user interface for the framework is using (#41b1e1) and would be very tedious to change; it would be nice to keep this color.

# Aesthetic:

We are trying to display that it is “from” this day and age. It should be sleek, minimalistic, and picturesque (most likely in the form of vector-based icons). A good example would be [Windows 8 Metro UI](http://i.imgur.com/G3cgOnw.jpg). It should be very “modular based.” The idea behind this is to allow customization of the layout. It should be fluid and simple.

# General:

The service is software based which will give you a lot of freedom as to how you design it. Do not feel constricted to the “web” mindset, animation (preferably simple) and scaling is all possible and encouraged. Users have been constricted to very static pages and any simple things like fading in and out content will bring a “wow” factor. You have a lot of freedom with the direction of the service. If you can think it, chances are we can implement it. Although, you should be reasonable with your designs (no crazy animation, 3D objects, anything that would bog the service down in terms of speed).

# Overview:

The following link contains a current overview of the project: <http://imgur.com/a/xXJaj#0>

The current constraints are 800x600 minimum with no set maximum (hence the rescaling). Any image resources should be vector-based as pixel-based graphics will distort and blur upon enlarging. The service will also be divided into a “User-End” service and an “Employer-End” service. We are currently working on the user-end as it is much larger and will define the service. You won’t need to worry about the employer-end for a while (or until you finish all of the user-end materials).

# Sections (Ordered in Importance):

1. Logo
   1. Preferably vector-based, but if anything, a high resolution bitmap.
2. [Login](http://imgur.com/a/xXJaj#0) 
   1. Standard login items, email & password box, remember email button, forgot password button, login button, sign up button
3. [Job Listings](http://imgur.com/a/xXJaj#5)
   1. Area to set filters and search criteria
   2. The actual area for all of the matched listings
   3. The job application
   4. Note: none of this has to be on separate pages
4. [Profile](http://imgur.com/a/xXJaj#9)
   1. This is the section where users will fill out all of their information that might pertain to a resume (Education, work experience, awards, etc)
   2. Basically a bunch of forms you need to fill out, not sure on how we can make this intuitive.
   3. Note: Some sections will need to be reusable (hence the modulation from above) because users will have multiple high schools / jobs (see how the high school “item” is exactly the same)
5. [Resume](http://blogs.msdn.com/blogfiles/chkoenig/WindowsLiveWriter/ExploringWPFDatabinding_BF1E/image_2.png)
   1. The current functionality is that the resume will auto-build itself and adhere to a strict layout.
   2. Each section of the resume will be able to be restricted to the employer so that the employer will have to request for further information.
   3. Ex. Education
      1. UNLV Undergraduate
         1. Majored in Computer Science
      2. MIT Graduate
         1. Majored in Computer Forensics
   4. Essentially a multi-expanding list was the current idea (the link demonstrates the concept, but obviously would need to be a lot more nice)
6. [Home Page](http://imgur.com/a/xXJaj#4)
   1. It needs to contain buttons to (I believe) 9 sections of the service
   2. It should contain a section of the user’s “Hot Picks” (basically high-matching job listings)
   3. Essentially a centralized hub, the link is probably a terrible reference of what we’re going for since it’s so basic in its current state.
7. [Messaging](http://vsee.com/static/hPKsDyL3W15fPvoAnQnXuHwsBR8eGU9WRpXfLSGMEjF.png)
   1. The service features the ability to schedule and perform online interviews.
   2. This includes something like only-chat, or the inclusion of video, with the possibility of maybe doing group interviews.
   3. A good example of this is Skype (linked) but without the contact list, and a chat box
8. [Calendar](https://lh6.googleusercontent.com/aJetsFMWLOqJb5vXeNPA4XUaof-F_xsPq4CkupR9RqlUq0W_yktmlogSmXowGyUyRfVn9ZKt=s640-h400-e365)
   1. The service features a fully-fledged calendar system
   2. There’s no real curveball here, Google Calendar (linked) is probably the best example. It’s up to you if you want to make it intuitive or something.
9. Settings
   1. Standard settings page.
10. [Notifications](http://ryanspoon.com/blog/wp-content/uploads/facebook-notifications.png)
    1. There should be a dedicated page for the list of notifications, but if you want to display the alerts somewhere that is up to you (think Facebook’s little globe thing on their navigation bar)
11. Classifieds
    1. This section is essentially suppose to act like the classifieds section of a newspaper.
    2. It’s merely an aggregator of all the listings we find that aren’t exactly directly connected to us.
12. Community Listings
    1. Behaves almost identical to the Job Listings but is focused around Community Service and things like that.
    2. The only thing that might look differently is the actual application process, where it might simply just be an RSVP

# Things to Consider (Not on the outline):

There is currently no navigation bar. We would probably need to include one as a way of navigating back and forth and maybe add user-defined buttons (notifications, job listings, etc). The service itself (the software) will need a way to present feedback to the user. Basically a notification, or an “alert” would be the better word, but separate from the “Notifications.” It shouldn’t interrupt the program at all. The service itself will also need to have a way to [prompt for confirmation](http://www.telerik.com/help/aspnet-ajax/media/grd_ConfirmationDialog.png) (whether this is a simple message box or something else is up to you). An important aspect to note is that the service will also ask the user to answer questions that will help us match the user. These questions will be asked on a semi-frequent basis 1 at a time (1 per hour, 1 per day, etc). Some ideas is to have a separate page for it and notify through the notifications, or simply have some sort of popup. The user will also need to go through an initial set of questions upon signing up for the service, this will need to be a dedicated section. The things to consider fall below (F. Home Page) and above (G. Messaging) in importance.

In addition, on the employer’s version of the software will be two specific pages not on the user’s edition: a form for employers to search for applicants currently looking for work, and a section for employers to set up their job listing. The first form will look similar to the user edition’s “Job Listings” page, while the second form will just be a set of fields asking the employer for information about the type of applicant they are looking to attract and hire (similar to the Job Listings page, except instead of looking at the information, this is where the employer is writing in all the information). In this second section the employer will be able to create/edit/delete a job listing, look at the current applicants for the job listing, filter/delete/”favorite” job applicants for the listing, and then interact with the job applicants by looking at their resume and sending them messages.

# Critical Items and Time Frames:

Items A-E is of upmost importance and will be necessary to have completed when we meet with our first investor. Our first investor meeting is in March and depending on how complicated your designs are, the implementation could take a few days to a week. F would be nice to have, but not absolutely necessary. Everything else can come ASAP and is of no rush.