Web Hosting Specifications

**Location**

It seems reasonable to choose a location within the UK, for legal and reliability purposes.

**Provider**

There are several well respected dedicated hosting providers in the UK, such as FastHosts, Rackspace and Heart Internet. They should provide Tier 3 or 4 data centres.

**Server**

The type of server used to host this service will depend on the performance and security level required. It seems that uptime, and security are major factors in choosing the hosting service for this project. There seem to be many opinions on whether it is best to use a dedicated server, or cloud services. There has been a lot of talk in the past about the security vulnerabilities of cloud services, but it is in the providers’ best interests to ensure a very high level of security and reliability on their servers, and the situation is improving. The cloud services offer high availability, and redundancy, and this can reduce the costs and hassle of maintaining a server yourself.

One of the advantages of a dedicated server is that modifications can be made to the configuration, to best suit your purposes. A client can install software of his own choosing at any time, as he controls the server entirely. Cloud based services are unlikely to be this flexible.

An advantage of cloud based servers, are that they charge by the amount of disk space and cpu time used, so they can start off quite cheap, and the costs will only increase, if the application becomes more widely used.

**Hardware requirements for dedicated server**

Processor

A single processor, with dual core should be sufficient. The dual core will increase performance, as we will have a SQL server, along with the web server on the same machine.

Memory

I would think that 16GB would be enough for now. This can be increased later, if we find that more is needed.

Hard disk storage

250GB would be more than enough for a web server. I would suggest having dual 250GB drives, in a RAID 1 configuration.  With RAID 1, the primary disk is mirrored to the second disk, so that a hardware failure in one disk would not result in loss of data. It can also speed up performance, depending on the implementation (I can speak to the hosting company about this).

Operating system

Windows 2008 Standard edition, with IIS , SQL server 2012 express.  The express edition of SQL server is free, and will work for databases of up to 4GB, which is actually pretty big for a database. This should be more than adequate for the initial phases of the deployment. When the user base increases significantly, then it will be necessary to purchase a SQL server licence.

**Management**

This can either be provided by the hosting company, or by a third person. The true level of management will probably vary from provider to provider. It would be good to get reviews from existing clients if possible on this issue.

Astutium have stated that they provide a fully managed service, but making changes to the server may involve a ticketing system, and there may be a time lag involved. They should make their management parameters clear in their paperwork. Some companies, such as Rackspace claim to have ‘Fanatical Support’, and will be on hand 24/7 365 days a year.

It will be important to have the hosting company configure the IIS server, as this can be quite involved. SQL server administration is more straightforward.

**Platform**

This should be a standard set up – Windows 2008 Server Standard edition, .NET framework 4.5, with IIS 7, and SQL server 2012 Express.

**Security**

The application will be using the .NET membership providers, which will restrict access to pages as required, through the use of user roles. Obviously, all users will have their own username and password (which should be of a minimum standard – at least 8 chars long, with upper and lower case chars, 1 numeric and one non alpha char.). Client names are encrypted, and users’ password will be encrypted in the database. Other data fields can be encrypted, as deemed necessary.

There will be further security provisions in the application, such as measures to stop SQL injection attacks, cross site scripting and protections against other code vulnerabilities.

An SSL certificate will be installed on the server, which will encrypt data being transmitted from the client machine to the server.

As the data is being stored in the database, with encrypted names, it should not be necessary to use encrypted hard drives. These will degrade performance, and make maintenance of the server harder. If it does become necessary to store documents with client data on the server, then an encrypted document folder can be created for this purpose.

**Backups**

Backups should be performed daily (or nightly), and should include databases, and documents. The databases can also be backed up continually, using log shipping, or replication to another server. This ensures that data can be recovered right up to the point of time where services or hardware failed. This backup plan will need to be maintained by the hosting service.

If all data is stored in the database, then the size of the backups should remain fairly small (less than 10Gb), and should cause no problems with disk space.

**Service Level Agreement**

1. Intellectual Property

As stated, you will retain all rights over intellectual property and data.

1. Instructions

As stated by the technician at Astutium, it is unlikely that any hosting company would agree to the need for written consent before making changes to the web server. The agreement will be made up front, and they will need to have access to the server to make upgrades to software, and perform maintenance as required. In the case of a dedicated server, it is not likely that the hosting company would make changes that would hinder your ability to serve your application.

1. Uptime Guarantee

Hosting companies make various promises regarding uptime. Rackspace make a guarantee of 100% uptime, but I cannot see how this can be ensured. Computer systems are highly complex, and can all fail in some way over their lifetime. The idea is to use multiple redundancy in hardware (Hard drives, PSU, cooling etc.), and keep the software up to date with latest patches to minimise the down time as much as possible.

It is important to get assurances from the hosting company for their guaranteed uptime, and ask them to explain how they will achieve this.

1. Compensation

This will depend entirely on the type of contract you enter in to with the hosting provider.

**Messaging**

1. Auto responders and Mailing list management programs

These services can be provided by dedicated third party companies, such as Mail Chimp. The pricing can be very reasonable from these large providers, and Mail chimp allows you to use their service free of charge if you are emailing to less than 2000 subscribers (12000 emails per month). This should be more than adequate during the early phases of deployment.

1. POP3 and Email services

These come as standard on all web servers, so that emails can be sent quite easily from the application.

1. SMS

SMS messaging services can be added in to a web application through the use of a DLL library. These services can be obtained through third party providers, such as Intellisoft. They are very easy to develop with, and the SMS credits can be purchased in blocks at about 6p per message.

**Data Transfer**

FTP Access, and anonymous FTP are provided as standard on any web hosting service. A dedicated windows server would also provide other methods of remote access, such as Remote Desktop and SSH.

The 3000Gb monthly as quoted by Serverspace would easily be enough for a service that has less than 100 concurrent users.

**Domain Hosting**

In terms of user access, it would be best, as much as possible, to integrate the various elements of the applications into one larger application. This will allow for greater control over roles, and will reduce unnecessary complexity. Portal front-ends can be built into the main application, and Content Management features can be added, to allow admin users to make changes to text, without having to understand web page design.

**Scale and Scalability**

Web Servers are designed to handle thousands of concurrent users, and large numbers of database transactions per second. Given the estimated initial load, it can be assumed that any modern dedicated server, or cloud system would easily be able to meet the requirements. As stated earlier, if cloud based systems are chosen, then the CPU, disk space allocation, and bandwidth can all be ramped up as demand increases.

**Testing**

It is a good idea to get a testing plan in place as early as possible. Testing with live data will quickly show up any validation problems, or database schema glitches.

Real users tend to enter data in a style of their own, and this can often result in ‘dirty’ data – especially in address fields. It is a good idea to build in validation checks, and data integrity checks right from the outset. These will give a good foundation for expanding the application in future.

Another feature that can be added to an application to reduce the debugging time, is application error logging (ELMAH or Log4NET for example). These work in the background, and will log all errors that occur in an application automatically during testing, ensuring that the developer has a good trace of any errors that occurred.

A bug database should be used to allow testers to report performance, and feature problems back to the developer. Only the tester should be allowed to sign off any bug fixes.

Use of source code control, such as Team Foundation Server is a good idea for version control, and collaboration. This may become necessary in future.