Response to the SUI Requirements document.

The table has following the parts:
- The requirement number.
- The category of my response.
  - I am either concerned about the requirement in general. Such as it should not be here or it does not make sense.
  - An estimate on a scale of 1 to 5 of how much work it might be.
- My more detailed comment on the requirement.

#	Category	Comment
3.4		Functionality Without Downloading: We need to identify specifically what the limitations might be here.
3.5		VO: SUI is required to define VO? I thought that was NCSA? Who has the responsibility to which which VO protocols we support?
3.7	concern	Title: Program-Friendly Interfaces? It means API? right? Not sure this requirement actually says anything.
4.5.1		API: Our packaging will need to be started, monitored, and requested via API. Not a hard requirement but we do not support that now
4.8	concern	Provenance: I do not believe that this is possible. It is far, far more complicated than it sound. We should work to remove this requirement.
4.9.*		Query Interface: None of this is hard as long as the database has it supported. This is really a DB requirement.
4.10	concern	Do we want an SQL box in the UI? I guess we should have query page where it is an option. The problem is we might not be sure what type of data is coming back. I don't think we can play down the concept of a power user. This clearly needs more definition
4.1.1		4 Output formats: JSON, VOTable, CSV, IPAC Table
4.14	concern	Who provides the dynamic cross-match algorithm. Pipe line? This should also be a DB and Pipeline requirement?
4.15.1		From the UI perspective everything is background. a Query should not block.
4.15.4		Size Limitation: is this pre-packaged or post-packaged (zipped)? We might not know the results set size until all the queries have been run and delivered the product to a staging area.
4.15.5		This is a DB requirement.
4.15.10		System Busy Indication: Very hard to get right. Not much code but very hard to define and recognize accurately.
4.15	concern	SUMMARY: I think the SUI performance requirements get it wrong. They are DB requirements. They are the type requirements you would put on static web pages not an AJAX web app. These should be rethought.
5.1	work: 2	Uncertainties: Our XY plots will have to handle uncertainties. They do do that now.

#	Category	Comment
5.3	work: 2	Multiple-resolution image displays: If the input to display is list of fits files at various resolutions then we don't yet have a fits viewer that takes more than one input fits file. This would be some work to support.
5.3.2	work: 5	We don't have anything in house to support this. I have seen web based tools that do it.
5.4		Mosaicking & coadding: this is really a pipeline or butler function.
5.5		Quick Look Image: I think producing low-res fits files would be better here. We can do more with them.
5.6.3	work: 3	quick-look aperture photometry using an aperture and annulus: We don't do this yet
5.7.*	work: 3	Time Domain visualization: We have a lot of the pieces but we will still have work todo beyond just putting them together.
5.9		False-color image display: I think this means 3 color plots
5.11.*	work: 2	SSO: We do SSO tracks but I don't have this set up in a general enough way. However, the hard technical problems are solved. We know how to get the information, we know how to plot tracks. There is probably a small amount of work here.
5.12	work: 2	Periodograms: We don't do this now but I don't think it is a big effort
5.13	work: 3	Light curve visualization: We have all the pieces, we need to put them together.
5.15	work: 1	Ancillary Information display: we can do all of this, we need to package it together
5.16	work: 1	Linked plot sets: I think we do this already, It might take a little updated to get all the features
5.18	work: 5	3D/4D data visualization: We don't have this capability. We will need to bring in WebGL
5.19.2	work: 2	Histograms: We don't do histograms. Would not be hard to add.
6.5.*, 6.9.* 6.10.*	work: 4	Workspace: This is a major feature but it not hard to do. From the client side I have not concerns. The big unknown is who is going server component to the workspace like IRSA (Walter) is doing. Who will write this? SUI or SLAC? *Note*: If I were to rewrite the documents I would combine 6.5, 6.9 and 6.10
6.6.2	concern	Storage Quotas: I don't think this is a SUI issue
6.7	work: 3	Scriptable operation: I think this is the idea of writing python wrappers around the HTTP API's
7	work: 5	Alerts: This probably should be written as a completely separate web app from the other parts of the system. It will be a significant effort. While the firefly package will provide framework to the web app, it still involves significant development that is different than anything else we have.
9	concern	Computational Tools: I don't really think this is an SUI requirement. I think this is more a pipeline requirement that the SUI might interface with.