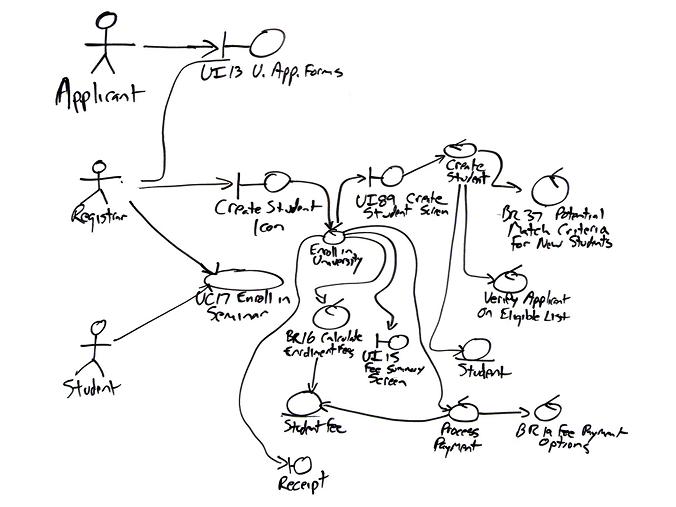
# Iconix

* It’s about understanding and documenting the user’s behaviour requirements, rooting out ambiguity in these requirements, and then using them to drive a good clean object orientated design.
* Iconix uses robustness analysis, which is essentially a slightly altered UML diagram with a low-level view which shows how end-user behaviour from various actors have been taken into consideration (example) <https://www.red-gate.com/simple-talk/opinion/opinion-pieces/agile-development-and-iconix/>



* Behaviour Driven Development, its development was derived from Test-Driven Development in order to answer questions such as “Where to start the process?”, “What to test and what not to test”, “What to call the tests?”, it enables business-readable language which allows you to describe a system’s behaviour without explaining how that behaviour is implemented. <https://medium.com/@TechMagic/get-started-with-behavior-driven-development-ecdca40e827b>