During Assignment two our group strove hard to avoid and correct the errors made and shortcuts taken during the previous assignment. Our first and probably most significant difficulty was, frustratingly at the very start of the process – what software are we going to develop? What is it for? Who is it for? After some indecision we eventually settled on the marking/assessment manager for schools. The only other reasonable concepts were for games, something it was felt would not be an appropriate project despite the scale as it involves focus on many things unrelated to the actual development of software. As the concept we chose to develop was my own this lead to an increased reliance on my knowledge of the concept. Something that, in retrospect, I think we could have better incorporated into our group structure and overall process.

As far as our initial planning, definition of methodology and so on, I believe the group produced good ideas, although these are clearly not enhanced with any large amount of real experience in actual software development. Having defined our problem and very roughly how we will approach it (our basic architecture, programming languages and similar tools) we defined our own development methodology and principles using a combination of our own preferences and perceptions of best practices. Our processes were really quite simple- something we wanted to and knew we could work within effectively. Were we to have our time over again I for one would like to try and contrast the same project using a widely known methodology such as Scrum or Extreme Programming rather than getting hung up in the details of defining our own processes. However, I do not believe this would be necessarily better as it may be that these methodologies do not suit the problem at hand or do not suit the team.

Despite clearly defining a set of processes and principles of our own, we still had trouble working within these. Our intention was to use a wiki for collating our notes from meetings and most of our artefacts (source code excluded) – we tended not to, keeping notes in all manner of other ways and occasionally not at all. Similarly we underutilised our git repository, not making full use of the comprehensive documentation facilities available. Using git itself was often a problem and going further into the project, I for one would want to investigate alternative source control and sharing methods. Failing that, once everyone had learnt more about using git, problems would hopefully solve themselves.

There were occasions where I simply had to admit to ignorance and inexperience and given the nature of the subject of software development, this was surely unavoidable. While, in the abstract we have made good progress, areas involving concrete numbers, especially laying out a schedule proved enormous hurdles. With minimal prior experience at the scale of this project, we had little comparison material to estimate how long things would take and resorted to (slightly educated) guessing. The important thing was we had some schedule from the start – which surely reflects the reality of the situation better than a blank page.

Of perhaps a more metaphysical nature is the necessary warping of our approach to, motivations during the project due to its true nature of a university assignment – not a genuine professional software development effort. While the intention was undoubtedly act as if it were, it was clear that it wasn't at all times, especially during the writing of this report. Most importantly, the student lifestyle did not necessarily accommodate the day to day schedule demanded by our methodology with tests, assignments and other such daily timetable fluctuations interfering with our intentions of regular meetings at set times during the week. Replicating a realistic environment for this

development process was easily the hardest part of the task. We could have redefined our processes to be extremely flexible around such matters but this seemed to directly conflict with principles of effective communication and quality control. It would have encouraged typical student procrastination; something that had to be constantly fought both individually and as a group – an important lesson learned from the first assignment.