Individual Retrospective

# My Details

Name: Jordan Litsas  
Degree: Master of Information Technology – Software and Services Development  
Company Name: Ontrack  
Role title or area: Backend Developer  
Target grade: Credit

# Self-Assessment

**Grade:** Pass | Credit | Distinction | High Distinction

The overall team progress and my individual progress has been slower than anticipated. For the first six weeks we as a team were under the impression that we would be building upon an existing code base for the conversion of jupyter files to PDF. At the six-week mark we had still not received any existing work and so the team met with Andrew Cain to discuss how best to move forward. We were advised that the existing code was not suitable and that we should start anew. Additionally, it was advised that we also add a feature to convert .docx files into .pdf. This is the task that I have taken responsibility for.

The following items include what have been completed to date:

1. Overall design for my feature contribution to outline the deliverables.
2. Research into how files can be transferred between docker containers, and a corresponding flow diagram to illustrate that to my team.
3. A program created with Java and the Apache POI API that converts a given .docx into a .pdf file.
4. Worked with Lachlan Foy to create a mock Ruby on Rails application to test our two docker converter apps in simpler environment than the existing OnTrack project. This way we can test how to incorporate multiple docker containers within a single application and have them communicate.

The following items are what needs to be done to finalise the feature:

1. Further research into how the file will be transferred from the user’s upload into the new docker responsible for converting the file, and then how to return that file to the OnTrack container.
2. Create a docker image for the .docx to .pdf application.
3. Bash script to initiate the docker container, and transfer/return the .docx and .pdf files.
4. Documentation describing the application and its integration.
5. Determining where the container should be allocated in the existing project.

This responsibility was not assigned to myself, I offered to take upon these duties. Andrew Cain suggested that we additionally provide a feature to convert .docx files, initially the project scope was limited to .ipynb. It was determined that the Apache POI library be used for this purpose, and given my experience with Java relative to my team members I decided it was best for myself to contribute in this area.

# Course Learning Outcomes

**Discipline-specific knowledge and capabilities**

*Develop a broad, coherent knowledge of the IT discipline, including its dynamic environment, with expert knowledge of the technological aspects of IT, and in depth knowledge in the chosen area of specialisation. Design, develop and implement advanced IT systems and software, and associated policies and procedures for optimal use and apply industry standards and best practice in one or more specialised areas of IT.*

This unit has especially prompted the breadth of my understanding within the IT discipline. Ruby, Ruby on Rails and Docker were all technologies I have none or very limited experience in at the beginning of this unit. As the unit has progressed, I am able to create and deploy my own Docker containers, become familiar with docker compose which will become vital as the feature develops given the multi-container nature of the application, can understand the existing codebase written in Ruby, and have the skills sufficient to contribute to that codebase. One problem I am currently experiencing is configuring my maven java application with docker. When I gather the knowledge to solve this problem, within this unit I would have taken on a new technology and incorporated my existing skills to provide functionality to OnTrack. By doing so I will further broaden my knowledge and engage with IT’s ‘dynamic environment’.

My docker skills have developed via reading documentation and creating demo containers from existing apps I’ve made. I’ve also learnt syntax and basic principles of ruby with the Ruby Koans interactive tutorial. Once my application is complete and properly integrated with the existing OnTrack codebase, I believe this should highlight the progress in my ‘software and services development’ skill base by nature of the project. Furthermore, given Docker is the industry standard container runtime, when I properly configure my app’s docker image it should illustrate my ability to learn and adapt to industry standard technologies and best practises. Lastly, I have begun to design how our team’s applications will communicate in its multi-container nature but utilising bash scripts. This shows my ability to bring together different concepts, Shell scripts and docker, and can be found at *figure 1.*

**Communication**

*Communicate effectively in order to design, evaluate and respond to advances in technology, future trends and industry standards and utilise a range of verbal, graphical and written forms, customised for diverse audiences including specialist and non- specialist clients, colleagues and industry personnel.*

When communicating concepts or system designs, I typically create a visual diagram to define data and communication flow with Lucidcharts. When working in a team, I firstly provide a written or verbal description of what I am about to design to provide feedback and gain approval. Lastly, when my task is finished and my design has been implemented, I will finish by generating a written document describing the task’s functionality and join it with the diagram. This is generally shared with the team on Trello and will be published to Github when our feature makes further progression. This addresses part of the learning outcome, namely communication with colleagues and industry personal. I have no experience communicating with specialist clients. In the remainder of the unit, I will engage with company leaders to get feedback from my documentation. The main example of responding to changing trends and standards for me within this unit has been the requirement to familiarise myself with docker’s terminology. Being able to solve issues with my team members was underpinned by my understanding of images, containers, how docker works etc. Overall, this is not an area I have had too much experience in besides generating documents, illustrating diagrams and communicating these to my peers.

**Digital Literacy**

Utilise a range of digital technologies and information sources to discover, select, analyse, synthesise, evaluate, critique and disseminate both technical and professional information.

This degree has exposed me to multiple new technologies to facilitate handling information. Firstly, Trello is likely the most prevalent websites I’ve used. Mainly to communicate, plan and synthesise group tasks with my teammates. It has been imperative to any group project I’ve participated in and I believe I can now use it at a professional standard. Secondly, a readme on Github has also been vital. I’ve taken upon advice from my previous lecturer Alessio Bontio that a repository without a readme is unprofessional and limits the usability of your code. I’ve followed this advice and believe I hit upon the key purposes of a readme on all projects I publish on Github. Thirdly, as mentioned above, I use Lucidcharts to outline any designs I have for a component/system/feature etc.. I completed a short Udemy course on creating software designs with such at an industry standard and believe by the end of my degree I should be able to complete this at a professional level. Lastly, I’ve had to learn many new technologies, such as Docker, Apache POI (an API that converts file types to and from those within the Microsoft Office Suite), Maven, Ruby and Ruby on Rails. In addition, I needed to learn these new skills in a short period of time, too short a period to spend watching tutorials and classes on Youtube. Therefore, I’ve had to expand on my ability to read technical documentation on the websites belonging to those technologies. Although at times I still need verbal explanations on how some technologies work, I’ve become far more proficient at disseminating technical documents for APIs and processes.

**Critical Thinking**

*Appraise complex information using critical and analytical thinking and judgement to identify problems, analyse user requirements and propose appropriate and innovative solutions.*

At the beginning of this degree I had minimal experience in developing software, and to be frank required a lot of hand holding to get anything working during my Introduction to Object Oriented Programming class in my undergraduate degree. Since then, I’ve been able to shift my way of thinking to be more aligned with how to develop software both in a practical and efficient way. During this unit, and as a junior last trimester, I have been required to analyse software requirements, and how to properly implement those. For example, I’ve had to understand the existing code base, dissect Docker implementation, and know how to integrate my feature accordingly. Although I have a long way to go in this regard before my degree is finished, I believe I am on track.

**Problem Solving**

*Generate IT solutions through the application of specialised theoretical constructs, expert skills and critical analysis to real-world, ill-defined problems to develop appropriate and innovative IT solutions.*

This unit has not been particularly relevant to this learning objective as the feature I am charged with developing, to convert .docx to .pdf, is clearly defined. Rather, I would point to my previous capstone unit where we were developing an investment application. The only guidance we were given was that it should focus on green/ethical ASX listed companies. My role was to develop an API, and I decided it should provide most of the features a typical investment application does. Firstly, the theoretical construct that was used was the Model View Controller architectural scheme. As most applications I’ve developed at university have used this system, I have become relatively adept with it. I’ve received feedback from tutors that my APIs are of a professional, and so I believe I have done quite well in this regard. The procedure I went through to boil down the problem was to both speak to the product owner, analyse competitor’s applications, and put my solutions to the team. Overall, I need to improve in this learning objective to ensure that my solutions are efficient, however I believe I am on track.

**Self-management**

*Take personal, professional and social responsibility within changing national and international professional IT contexts to develop autonomy as researchers and evaluate own performance for continuing professional development. Work autonomously and responsibly to create solutions to new situations and actively apply knowledge of theoretical constructs and methodologies to make informed decisions.*

So far in my degree I’ve been exposed to technical theories and skills to develop software, and my engagement in professional IT contexts and research has been limited. What points I can put towards this learning outcome achievement is that each project I’ve worked on has been novel, in that I have not had previous experience in the project goal or at times the technologies used. However, my knowledge of theoretical constructs such as object-oriented programming design principles and architectural techniques such as MVC have been at the forefront of my design decisions. This is my weakest learning outcome; however, I hope to be able to improve upon in it during SIT740 – Research and Development in Information Technology.

**Teamwork**

*Work independently and collaboratively towards achieving the outcomes of a group project, thereby demonstrating interpersonal skills including the ability to brainstorm, negotiate, resolve conflicts, manage difficult and awkward conversations, provide constructive feedback, and demonstrate the ability to function effectively in diverse professional, social and cultural contexts.*

As opposed to ‘self-management’, my progress towards qualifying myself in this learning outcome has been the strongest. This in part is due to these capstone units. Before these units, all my projects were individual ones, and so all my achievement for this outcome has been done in the past trimester and currently. Obviously, all the points listed in the learning outcome are highly relevant for these units. I have had no interpersonal issues or teamwork issues that have hindered my development progress, and so I consider my efforts so far to be a success. One point I will need to continue to improve upon is to ‘provide constructive feedback’ in areas where my technology expertise is lacking, however this should naturally be mitigated as my education continues.

**Global citizenship**

*Engage in professional and ethical behaviour in the design, development and management of IT systems, in the global context, in collaboration with diverse communities and cultures.*

Fortunately, I’ve had the pleasure of working with many of my peers in China and India. Navigating language and cultural barriers has been relatively easy given I grew up in an extremely diverse community, and so I have not had to improve upon this during this degree but rather believe this is already at a professional level. Otherwise, during the development of my ASX company data tracker, it was important to display to the user that the prices of shares shown were 15 minutes out of date. This is because the MVP we developed had no budget and 15 minutes was to shortest timeframe available. This decision was made to ensure users did not make purchases with poor information that could have drastic implications for their financial success. Although I do not have any other examples of this learning outcome particularly because I have not had exposure to such scenarios, I believe when I graduate I will be fully qualified in this area.

# Overview of Evidence Collected

***Note:*** The vast majority of mine and my team’s work has been discussions on how we should proceed. As mentioned earlier, up until week 6 we were under the impression that there would be existing code for us to improve upon – Andrew Cain has informed us that this is incorrect and Jordan Trainor our Team Leader had informed us the opposite (hence our confusion). I am aware that my contributions are minimal, but I hope you can take into consideration our challenges. Even though there is little work here to present, my application is functional and once a docker image is created (within the next week) will be ready to be integrated into OnTrack.

**.docx to .pdf converter – Maven/Java/Docker (tbc.) application**

Resource: <https://github.com/thoth-tech/jupyter-notebook-prototype/tree/main>

* I developed a java application that converts all .docx files to .pdf. Files are consumed within the ‘input’ folder and the .pdf is placed in the ‘output’ folder.
* There have been issues making a docker image for the application, and more research into Docker and Maven is required.
* The application works as expected on files of differing lengths and content (images, text, graphs etc.).
* The application has taken 20 hours so far to research, design and develop.
* Those within ‘docxtopdf’ is all my work, and the rest has been completed by Lachlan Foy.
* Once the app image is made, Lachlan and I will integrate our work into a Ruby on Rails prototype to ensure a proof of concept, and finally integrated into OnTrack.

I’ve included this repository as it represents the majority of my contributions. Although it was not a part of the initial feature requirements, as that was limited to jupyter notebook conversions to pdf, after meeting with Andrew Cain and explaining the team’s difficulties getting started he suggested a .docx conversion as well. The application works exactly as intended and fits into the design which is my next evidence below.

**Software process for converting files to pdf within docker containers**

Resource: <https://lucid.app/lucidchart/81841374-52fd-47ba-bc16-a1e2ce57ef34/edit?invitationId=inv_eed4a77d-9de5-4b29-85ec-01e3041b909d>

* After some research into how to transfer files between docker containers, this is the design that will represent my work within the next sprint.
* The design explains how docker commands and bash scripts will integrate OnTrack with my team’s submissions.
* The research took 5 hours and design took 30 minutes.
* I completed this by myself.

I’ve included this as it will be the structure for the following sprint, and outline how Lachlan and my work will actually integrate with the existing application. In this sense it is imperative to have this document.

# Reflections

**If you had to pick one thing you have done to best represent what you have achieved in capstone so far, what would it be and why?**

Teamwork – being able to overcome obstacles in differing scenarios both individually and within my team. For the first capstone (which was on a different project), I created an API with nodejs and several other financial data APIs. My frontend counterpart and I had to work quite closely, and abide by a data flow interface document, to ensure that our work was perfectly integrated. Given the language barrier (Wensong is a native Mandarin speaker and I English), we had to rely heavily on translation applications and visual diagrams. This in hindsight was actually very valuable as it required me to ensure that whatever it was that I was trying to communicate was simple without much abstraction. Therefore, I had to become an expert on the subject matter and my project. This skill has transferred onto the current capstone unit, as I am working with Java and Maven which I am not very experienced with, but still need to communicate my work with my team. I have handled with steep learning curve well, and my team is clear on what I have completed, what the challenges are, the timeline for completion, and how they could assist if they wanted to. I would point to my evidence to support these claims.

**What is the most important thing that you have learnt so far in the capstone program?**

My most important lesson is how to effectively read technical documentation and implement this new knowledge. Java, Maven, Docker and learning the existing OnTrack codebase are all things I have little or no experience in. And so, to be able to produce work with my team I’ve had to quickly gather working knowledge in these technologies. There are still gaps in my skills, but I’ve learnt enough to be able to produce working software.

**What do you hope to be able to achieve by the end of the trimester? How will this help you showcase your capabilities, and how will it help advance your company or one of the capstone cohorts?**

By the end of the trimester, I am aiming to be able to have the skills to enter a new team, quickly get acclimated with the project technologies and existing work, and effectively contribute without having to rely on teammate’s guidance. This should help me showcase my abilities when I join a new team, and I am able to promptly show that I can be a productive member and not a hindrance. To be able to achieve this, I will need to put this into action within the next sprint where I will integrate Lachlan and my work to OnTrack. In doing so, students who further extend OnTrack’s file conversion capabilities will have a clear and working codebase to work with.

**If you could go back and redo the start of the unit, is there anything you would do differently? How would changing this have helped improve things?**

The biggest thing I would change is my leadership and proactiveness. Yes, we faced significant delays and challenges in getting started. However, I believe it was not acceptable for us to be complacent, and rather I should have put to the team that we speak with Andrew Cain far earlier than we did. This would have given us several more weeks to progress our feature, improved my own communication and leadership skills, and made this retrospective far more compelling.

# Evidence

1. **System design:** <https://lucid.app/lucidchart/81841374-52fd-47ba-bc16-a1e2ce57ef34/edit?invitationId=inv_eed4a77d-9de5-4b29-85ec-01e3041b909d>
2. **Team repository:** <https://github.com/thoth-tech/jupyter-notebook-prototype/tree/main>
3. **Personal repository:** https://github.com/jordanlitsas/docx-to-pdf-converter
4. **Ruby upskilling:** <https://github.com/jordanlitsas/ontrack-ruby-learning>

.