

INFO1111: Computing 1A Professionalism

2023 Semester 1

Skills: Team Project Report

Submission number: 2

Github link: <https://github.com/hlia3350/INFO1111-CC25-02-GROUP.git>

Team Members:

Name	Student ID	Levels already achieved	Levels being attempted	Selected Major
LIANG, Haoyang	520040094	A	B	SW Development
LE, Duc Anh	530190891	A	A	Data Science
CHOW, Ryan	530517261	A	A	Computer Science
RICHARDSON, Callum	530510734	A	A	Cyber Security

Contents

1.	Teamwork	2
1.1.	Developing industry skills	2
1.2.	Submission 1 contribution overview	3
1.3.	Submission 2 contribution overview	3
2.	Level A: Basic Skills	4
2.1.	Skills for SW Development: LIANG, Haoyang	4
2.2.	Skills for Data Science: LE, Duc Anh	4
2.3.	Skills for Computer Science: CHOW, Ryan	5
2.4.	Skills for Cyber Security: RICHARDSON, Callum	6
3.	Level B: Tools	8
3.1.	Tools for SW Development: LIANG, Haoyang	8
3.2.	Tools for Data Science: LE, Duc Anh	9
3.3.	Tools for SW Development: CHOW, Ryan	9
3.4.	Tools for Cyber Security: RICHARDSON, Callum	9

1. Teamwork

1.1. Developing industry skills

Approaches to maintain continual learning: *Ordered from 1 being most effective and 5 being least effective*

1. **Finding an online course** that teaches you the topic of your choice. The ideal course has high ratings and a high completion rate, as this means there will most likely be a large community to help you and share resources. The course should also have quizzes/assessments so that you can test your knowledge. Udemy is a great resource for this as it allows you to set up weekly goals for what you wish to complete, as well as giving you a certificate at the end of the course which looks great in a portfolio. Lastly, online courses give you the chance to receive expert advice from the course instructor/s who often find time to respond to your questions.
2. **Practical application** One of the best ways to learn a new tool or programming language is to apply it to a real-world project. This could be a personal project, a contribution to an open-source project, or a project for work or school. By using the tool or language in a practical setting, you can gain experience with it and understand how it works in real-world scenarios. You will also be forced to solve real problems, which can be a great way to deepen your understanding and discover new features or techniques. While this approach can be challenging, it can also be very rewarding and provide a strong foundation for future learning.
3. **Cooperative programming** Cooperative programming is a collaborative learning method in which two programmers work on the same project or work on the same problem. Because of working together, you can or instant feedback, learn from each other and share useful knowledge or ideas. This keeps both motivated and engaged because someone is learning with you and from you. Find partners through online forums or communities, or connect with classmates or colleagues in your field. However, there are certain problems with this method. Cooperating together means that it is easy to eliminate people's desire to explore and lose the ability to work independently. Therefore, this method is not absolutely effective and beneficial.
4. **Working on open source projects** is a way to put your skills into action on a community project. People can provide you with guidance and resources, which will further boost your skills as you are learning from people who have already understood and learned these concepts. However, this is a double edged sword as learning from random other people instead of experts as it can lead to misunderstanding. This is also a rewarding way to demonstrate your skills and keep them in practice. This is not the most effective way of learning new skills, as you are mostly demonstrating your own knowledge, but you can discover holes in your own understanding by identifying things you do not yet know which is useful to further your skills.
5. **Social media** is a tool that can be used to keep yourself up to date with updates and current trends related to different pieces of software and applications. Reddit is an application that consists of different communities relating to a mass of topics, including Python, Java and Wireshark. Members can post questions or comments to the public, and people can respond to these. Some of these communities are owned by the company that make the software, which ensures the information is reliable. This is a very passive method of continual learning but can be useful if you are not in the right head space to "work" but wish to stay immersed in the

1.2. Submission 1 contribution overview

CALLUM: I believe that our team has worked together excellently. We have taken on a democratic approach and I feel that there is clear team efficacy. We have remembered our core values as a team and I believe that everyone feels that they are respected and that their opinions are heard. The work was divided evenly as each member had to write their own section, and since there were 5 learning approaches to consider and 4 of us, we were able to divide that up as well. For the future, I believe as a team we could make a time schedule so that we have deadlines to keep ourselves to.

Young: I think our teamwork has been excellent, our timing, and our overall planning so far have been excellent. Every week in each tutor class, we will gather together to discuss this week's lecture content. We have planned the whole group project in the last two weeks. First of all, each of us selected the relevant major, and we planned each section on the topic of teamwork. Everyone on our team is friendly and respects each other while efficiently scheduling and getting work done. Comments/feedback from each of our team members will be taken seriously. I believe that with such a team, we can achieve good results in any activity related to teamwork.

Steven: As we've neared the first submission date for the skills assessment, I feel as if our team has had a fair understanding of the group's core structure and individual strengths. Every one of our members has been able to compromise to reach a point of mutual respect and understanding to be able to achieve our goals for this project. We were able to navigate through common issues and personal goals with much ease and see to it that our project may be completed in a relatively timely manner. I believe that as we progress through the course if we maintain such flow, we could achieve our desired results without too many obstacles in the way.

Ryan: It is evident from my time with my team that all 4 of us have a solid understanding and have been working diligently towards completing the Skills assignment. My team's ability too proactively approach situations have allowed them to identify potential challenges within the assignment, and has been reflected in the quality of work I have read over. The effective communication and collaboration within the team through our Instagram group chat has also contributed to the project's success. My team has done an amazing job leading up the first submission date and despite a few obstacles that had stopped us for moments, I believe we have achieved the A levels we all desired to get.

1.3. Submission 2 contribution overview

Young: For the second submission, from the moment our team members know the submission time of the second submission, we have already started working. We first confirmed the level to be reached this time, and we assigned work. We confirmed that the two tools that everyone should mention in Level B are different, so as to avoid us choosing the same tool. At the same time, we help the team members to suggest some tools to choose for different majors. At the same time, we ensured that everyone follows the requirements of level B, that is, including screenshots of updated Latex, etc. I believe that with such an efficient, enthusiastic, and helpful team, we can also achieve our goals in Level B and future Levels.

2. Level A: Basic Skills

2.1. Skills for SW Development: LIANG, Haoyang

Strongest Skill:

At the current stage, the skill I am best at is ADEV[5], which is the animation development skill. This skill is very important for the profession of software development, almost one of the necessary skills for this profession. The importance of this skill for the profession of software development is as follows. First, animation development skills to create interactive and engaging UIs and experiences. This is an important aspect of software development that helps users to use the software more intuitive and engaging. Second, if you are engaged in game development, the animation in the game is very important. This skill can create realistic and believable animations for the characters and objects in the game. Effectively attract users and provide a real sense of experience. Third, animation can improve visual interest and participation. On the development website, excellent animation production can also allow users to get excellent feedback. Secondly, animation can also be used in APP development, especially in the interface design of APP. This skill is of great help to developers engaged in software program development. So far, I have successfully made a simple game and a simple company product display website to prove that I have made some progress in this technology and am good at it.

Weakest Skill:

At the current stage, the weakest skill i got is URCH[5], which is the user research skill. The skill of user research is a basic skill of software developers. Through this skill, software developers can better understand the needs and preferences of target users. This skill involves various data collection, analysis and synthesis methods. These It is very important to improve the user experience of the product. Proficiency in this technique leads to higher user satisfaction and engagement. These skills also help software developers improve software to meet user preferences and make more informed decisions about the overall product, which can guide decisions related to design, functionality, and other important aspects of software development. Bringing the skill of user research into the software development process can help reduce the risk of costly mistakes. Without user research, software developers can design features that they think are important but that users will find confusing or unnecessary. So by conducting user research, developers can avoid these potential problems that make software development more difficult and more expensive. Therefore, the skill of user research is very important for the major of Software development. Here's what I intend to do to improve this skill in the future. I intend to combine study and practice, and participate in seminars, or participate in online courses and read books to provide me with valuable theoretical knowledge and practical skills. And conduct research on real users by employing recruiting participants, conducting interviews, conducting surveys and analyzing data. I believe that by doing this I can improve my proficiency in this skill in the future.

2.2. Skills for Data Science: LE, Duc Anh

Strongest Skill:

The research skill (RSCH) [5] is currently my best skill relating to Data Science. Research skills are vital to data science as involve the act of collection, analysis, and interpretation of data. Good researchers can gather data that is both reliable and accurate, the data they collect should serve the purpose of providing insight for decision-making that would yield the desired results. The data collection must be relevant and useful, it serves the purpose

of giving context to the situation in which the researcher can apply an appropriate solution. Research skills don't only mean the act of gathering data, but also knowing what to do with that data and how it can affect the outcome of one's research. Having good research skills means understanding which solution can be applied to a given problem, drawing meaningful connections from one finding to another, and articulating those findings effectively. Data scientists are generally the first to know of emerging trends and patterns, a good researcher can provide a company with informed analysis and accurate prediction to gain leverage in a field. Throughout high school, I often involved myself in projects or activities involving the market analysis and human research. Many of the jobs that I did was related to selling a product to users in hopes of raising money for a cause. Though my activities were cut short by covid, I have participated enough to have hardwired a much-appreciated skill in my brain. I've been able to prove this skill in many of my classes relating to any type of research and real-world applications of said skills. Outside of class, I participated in charity events catered to students to rack up community service hours to graduate and build a decent portfolio. Many of those charity events involved selling products to people for sake of various charities. Though my achievements are few, I hope that develop my skills further through exposure until they can be applied in the workplace.

Weakest Skill:

Unfortunately, one of my weakest skills is in Data visualization (VISL)[5]. I believe the area where I lack is my understanding of visualization. Understanding data that may be redundant to show or which piece of information fits best with another to be able to precisely show a trend or correlation. As a data analyst, the whole point of the role is to be able to understand and convey the information that is being presented by the data. By visualizing connection and causation, the data analyst can accurately represent trends and correlation. Visualizing data is paramount to convey one's understanding of it and thus grant them the proper parameters to be able to make predictions and produce anything of value from said data. Proficiency in Data Visualization is what separates a great data scientist from a mediocre one. I have only recently developed an interest in studying data and could only wish to attain such a skill so early on in my studies. This project provides me the chance to measure my growth as I try to turn this weakness into one of my skill sets. Luckily, I was given a chance to take up a role in graph visualization for a different course. This could work in tandem with my current arrangements as it allows me to improve my skills for both courses. By introducing myself to different types of data and finding context to the issues at hand, I'll naturally find a way to make a connection and visualize it. I had very few technical skills before entering this major, though I feel that if I continue to challenge myself I could make up for what I lack

2.3. Skills for Computer Science: CHOW, Ryan

Strongest Skill:

Numerical analysis (NUAN)[5] has been by far my strongest skill that is heavily relevant to not only getting jobs into high status tech companies but as well helps in efficiently creating code. Having a strong background in numerical analysis is key to passing knowledge interviews from most notably the M.A.N.G.A (Meta, Amazon, Netflix, Google, Apple) companies, which consists of interviewees solving complex computing problems in the most efficient manner. Not only must programmers solve a set of these problems in their interview time, but they must also create them in the most efficient way's possible and as well explain their logic throughout, something that can only be accomplished with a strong background in numerical analysis. By understanding these techniques and how to solve

these problems efficiently, these programmers can excel in creating solutions in areas such as optimization, simulation, and modelling. Furthermore, by understanding these complex algorithms and data structures, programmers who are efficient in numerical analysis are much more aware of the logic of their code, thus being able to catch out errors from their own code efficiently. Understanding such complex logic also allows these programmers to understand the code of other programmers much faster than one with a shallow expertise in numerical analysis. This allows for more efficiency in terms of editing and adding to another's code, which is immensely helpful in the tech industry as it bolsters on team work and communication. My proof of my strength in NUAN is my experience with the website Leetcode, which focuses on solving complex programming problems, with right now having completed 76 problems, (40 Easy, 30 Medium and 6 Hard). This has allowed me to gain in depth knowledge of numerical analysis.

Weakest Skill:

Programming/software development (PROG)[5] is a skill that is integral to any job in the computer science field that I have no experience in whatsoever. The skill, described by SFIA as "Developing software components to deliver value to stakeholders." is key in maintaining any company and requires a moderate level of communication between not only clients but team members of your projects such that your final piece of software satisfies the needs of the client. By developing skills in PROG, programmers are able to fully understand the requirements that are needed for software development, allowing for more efficiently creating code as well as helping others understand or implement their code. By having a intricate understanding of client needs, computer science professionals are able to assess challenges and limitations that might arise during planning of new software, allowing for programs to be created with more efficiency by reducing any time that consists of hitting road blocks. The most effective way in improving PROG is too intern at software jobs too see how a software company functions. Through this I can learn how the software development cycle works and understand the most effective and inexpensive way of creating software for clientele. Furthermore, by being given projects I have to work on I can understand the intricacies of a project.

2.4. Skills for Cyber Security: RICHARDSON, Callum

Strongest Skill:

I believe my strongest skill relates to Governance[5], as defined in the SFIA. It is described as "Defining and operating a framework for making decisions, managing stakeholder relationships, and identifying legitimate authority.". This is essential for cyber security as a company is likely to have a team of people working on creating safe networks and other things. Teams must have some type of governance in order to be efficient and purposeful. Thus, governance is a vital skill for cyber security. I believe that I am a strong leader and I like creating structures that allow for constant and efficient work. I have experience in leadership through being a high school prefect, which gave me the chance to set up meetings and create a schedule/agenda for them. This translates to the working world as collaboration is an important part of work, and collaboration requires communication/meetings. These meetings also require structure so that they are useful and productive. I also pride myself in being able to consider all possibilities of an event, which will prove helpful in decision making and creating a framework where all is considered. This has stemmed from my experience as a surf life saver, where considering the dangers and coming up with potential rescues is vital for people's safety. I have never managed any kind of stakeholder meeting, but as stated I have experience in setting up and leading

meetings in high school.

Weakest Skill:

Information security[5] is obviously vital to cyber security, as it is in regards to Defining and operating a framework of security controls and security management strategies. This is a skill that I have identified as being weak as it is very specific, and I have not yet had a chance to master the technical skills required to confidently say I am an expert in information security. However, there are aspects of this skill that I can begin to master before I go into the work force. For instance, information security involves Performing basic risk assessments for small information systems. I have done a type of risk assessment in high school for science, so I understand the basic concept of identifying the cause, effect and prevention of a risk. I believe in order to further my understanding of information security I should begin with communication networks and understanding how data packets can be protected with things such as parity checks. Information security also involves understanding business strategies in order ensure compliance between the strategies and information security. I could strengthen my skills by reading up on different business strategies and explore case studies concerning conflicts between the two.

3. Level B: Tools

Level B focuses on exploration of key tools used within professional computing employment. All companies make use of a range of technologies and tools (often as part of a tech stack). These tools might be implementation languages; design tools; data analysis tools; collaboration technologies, etc. Each student should identify two tools that are widely used in industry and which relate to the major you are focusing on for this project. You should then describe:

1. The main functionality of those tools;
2. The ways in which those tools are used;
3. Any weaknesses or limitations of those tools.

As examples (which you shouldn't now use): Computer Science: eclipse; Software Development: github; Cyber Security: Wireshark; Data Science: Hadoop.

Note also that no two students in the same tutorial should choose the same tools, so your tutor will maintain a list of those that have already been selected. You should therefore check this list and then confirm your choice with your tutor prior to researching your proposed tools and spending time writing about them. (Target = ~200-400 words per tool).

Also, in order to achieve level B each student needs to be able to demonstrate capability with git and compilation of LaTeX documents from the command line. To demonstrate this, your team (or at least those members who are aiming to attempt level B) should do the following:

1. Select one member to:
 - (a) Create a local github repository for the project. This repository should contain the main LaTeX documents, as well as a subdirectory called "screengrabs";
 - (b) Create a repository on github for the project;
 - (c) Connect your local repository to the remote github repo;
 - (d) Push your local repository contents to the remote repo;
 - (e) Add all team members (and your tutor and unit coordinator) as members to the remote repo;
2. Each additional group member should then clone the remote repo;
3. Each member aiming to achieve level B should then be able to use the remote repo (and pushing and pulling changes) to demonstrate collaborative editing of the LaTeX documents.
4. And each member aiming to achieve level B should also do a screengrab (or multiple screengrabs) showing their local successful compilation, on the command line, of the final LaTeX document. This should be added to the screengrabs folder in your local repo and then pushed to the remote repo so that your tutor can view it.

3.1. Tools for SW Development: LIANG, Haoyang

First Tool: Eclipse IDE

Eclipse is an integrated development environment (IDE) primarily used for developing Java applications, though it also supports other programming languages and technologies

such as C/C++, PHP, Python, and web development[4]. Eclipse's main functionalities include code editing, debugging, compiling, code analysis, and version control. Additionally, Eclipse offers various plugins and extensions to enhance its functionalities and extensibility [6]. Developers can use Eclipse to create and manage Java projects with ease. Eclipse provides a robust code editor with automatic code completion, allowing developers to write efficient, standardized, and maintainable code [3]. Furthermore, Eclipse offers debugging tools and analysis tools that can help developers quickly locate and fix errors in their code. To use Eclipse, one can download it from its official website and install it. After installation, users can install relevant plugins and extensions, such as web development plugins and version control plugins, as needed [3]. While Eclipse is a powerful IDE, some of its limitations include its high memory consumption, slow startup speed, and occasional stability issues. Moreover, due to its extensive functionalities, Eclipse may have a steep learning curve for beginners [6]. In summary, Eclipse is a powerful IDE suitable for Java and other programming languages' development. Its stability and extensibility make it one of the top choices for developers.

Second Tool: Unity

Unity is a cross-platform game engine that can be used to develop 2D and 3D games, virtual reality, and augmented reality applications. Unity has robust functionalities and an extensive resource library, making it widely used in various fields such as gaming, education, architecture, and healthcare. Unity's main functionalities include scene editing, physics simulation, animation creation, material editing, and game logic scripting. Unity also provides several development tools such as Visual Studio Tools for Unity and Unity Profiler, which can help developers improve productivity and performance [7]. Additionally, Unity offers numerous plugins and extensions such as Vuforia, ARKit, and SteamVR that can assist developers in quickly implementing VR and AR applications[2]. Developers can write code using C Sharp or UnityScript in Unity and utilize Unity's editor and tools to design scenes, simulate physics, create animations, and edit materials. Furthermore, developers can use the Unity Asset Store to download and use various resources and plugins to increase productivity [2]. Unity's limitations include a steep learning curve, high memory usage, and stability issues in some versions[1]. Additionally, Unity's performance is limited by hardware and may require optimization. In summary, Unity is a powerful game engine and application development tool suitable for various fields.

3.2. Tools for Data Science: LE, Duc Anh

Your text goes here

3.3. Tools for SW Development: CHOW, Ryan

Your text goes here

3.4. Tools for Cyber Security: RICHARDSON, Callum

Your text goes here

Bibliography

- [1] Alex Cicala. United we develop: A breakdown of unity's strengths and weaknesses, 2016. See <https://blackshellmedia.com/2016/11/01/united-develop-breakdown-unitys-strengths-weaknesses/>.
- [2] Game Designing. Unity 3d games tutorials and beginners guide, 2021. See <https://www.gamedesigning.org/learn/unity/>.
- [3] javaTpoint. How to use eclipse for java, 2020. See <https://www.javatpoint.com/how-to-use-eclipse-for-java>.
- [4] Pooja Gupta. What is eclipse ide?, 2023. See <https://www.educba.com/what-is-eclipse-ide/>.
- [5] SFIA. All skills a-z, 2022. See <https://sfia-online.org/en/sfia-8/all-skills-a-z>.
- [6] Software Advice. Eclipse ide reviews, 2023. See <https://www.softwareadvice.com/app-development/eclipse-ide-profile/reviews/>.
- [7] İSMAIL ÇAMÖNÜ. What can you do with unity game engine?, 2020. See <https://www.codinblack.com/what-can-you-do-with-unity-game-engine/>.