**Testing Plan**

Firstly before testing the student result check system, we need to find what to test first and what to test most. I used the Risk Matrix to demonstrate the risks of multi-dimensions that will be hidden in our system. Then I partition the risk in three levels High, Mid, Low (H, M, and L). From the form, I find there are two high lever risks in our system, knowledge of area and impact on users. Also has two mid-level risks high traffic and how visible. 

The knowledge of the area is the knowledge for developers. We need to understand how this system works does. This could be a high risk in during the development, so we need to test every developer to confirm they understand the logic of the system. If not assume that will be big trouble for the future implementation. In our development group only with two peoples and we all understand the logic for the system but I still put this for high risk because we not sure we will have enough knowledge control for future development. Another high-risk level impacts on the user, if our system has a bug or error it will impact lots of users because students will use the system we need to test the function, performance and user interface as well. We will check the back end, and front end first because this part will spend lots of time. In the test we need to confirm the student can log in the system and see the result correctly, also can log out. After that, we need to test the registration function for each student. Finally, is the student registration information can store in the database correctly. Another part should test in the database in the back end. Make sure the administrator can insert data without errors. User interface part we also will test, but the current user interface is only available for pc web browser, we need to check the online system can fit most popular browser that user often uses.

There are also have two mid-level risks from my analysis in the form. I think High Traffic is one of the mid-level risks. The High Risk in this form does not mean the transport, that means the high usage rate for users. Lots of students will use the system to check the result. This is the high traffic for the system. If our system crash with a little bug with effect many students. From here we can see we need to force the testing on function, performance, the user interface in front-end and database in the back-end. Another mid-risk is how visible when the system has a bug if the bug is happening in the user interface the user will see the bug. This situation is not suitable for our project. If the bug is occurring in the back-end, the user cannot see the bug easily. This situation does not affect our reputation for our product.

All in all, this is the risk analysis depend on the risk matrix for the system. We also find some low-level risks but because the time management reason we cannot test all things, so we need to test the function, performance, user interface most and first because if those things are correct then the system can run correctly as well at the same time we also can keep a high usage rate for users. This is our basic testing plan for the current version of the system.