科技部

111年度大專學生研究計畫申請書

一、綜合資料：

| 申 請 人 ︻ 學 生 ︼ | 姓 名 |  | | 身分證  號　碼 |  |
| --- | --- | --- | --- | --- | --- |
| 就 讀 學 校、  科 系 及 年 級 |  | | 電　話 |  |
| 學 生 研 究  計 畫 名 稱 |  | | | |
| 研 究 期 間 | 自111年7月1日至112年2月底止，計8個月 | | | |
| 計 畫 歸 屬 司 別 | □自然司　　 　□工程司 　□生科司  □人文司(含科學教育領域) | | | |
| 研究學門代碼及名稱 |  | | | |
| 上年度曾執行本部大專學生研究計畫 | □是（計畫編號：M0ST － － － － ）  □否 | | | |
| 指 導 教 授 | 姓 名 |  | | 身分證 號　碼 |  |
| 服 務 機 構 及 科 系(所) |  | | | |
| 職 稱 |  | | 電　話 |  |
| 補 助 經 費 | 項 目 | 金 額 | 說 明 | | |
| 研 究 助 學 金 | 48,000元 | 每月補助研究助學金新臺幣6,000元  ，8個月計新臺幣48,000元。 | | |
| 耗材、物品、圖書及雜項費用 | 元 | 依研究計畫實際需求擇優補助，每一計畫最高以補助新臺幣20,000元為限。 | | |
| 合 計 | 元 |  | | |

表C801

科、系主管姓名：

（學生就讀學校）

二、研究計畫內容（以10頁為限）：

題目：基於DevOps之程式作業品質測試分析系統

1. 摘要

大多數的程式批改軟體，像是瘋狂程設、online judge等等，他們批改的要求只有程式執行的結果是否正確。沒有辦法針對程式的撰寫品質進行檢查，此情況容易造成學生只在乎結果是否正確，不在乎程式碼撰寫的品質。

因此我們希望能開發一個具有程式自動批改功能的教學系統，系統的功能除了擁有一般批改軟體有的單元測試來判斷答案對錯之外，會再加入檢測程式品質的功能，並且在檢測品質的同時提供使用者改善的建議，藉此讓使用者能根據系統給予的提示，提升程式碼撰寫的品質。根據Pulasthi Perera, Roshali Silva & Indika Perera的研究[2]，使用DevOps開發模式有助於提升軟體程式碼品質，所以我們決定將DevOps的開發模式套用在學生繳交作業的方式，在學生熟悉軟體開發流程的同時，也能提升撰寫之程式碼的品質。

關於程式品質的考量點，我們找了很多相關論文以及文章，最終決定採用CISQ(Consortium for Information & Software Quality)組織對於程式品質(Code Quality)的規範當中的安全性(Security)、可靠性(Reliability)以及可維護性(Maintainability)這三項指標，程式品質檢測的重點也會放在這裡。

為了能夠更方便、直觀地了解學生的學習狀況，我們會將檢測結果統整到儀表板上顯示出來，儀表版會擁有題目分數、程式碼品質檢測結果以及學生程式碼的品質曲線圖等資訊，並且系統能提供使用者能直接使用的建議程式碼。

1. 研究動機與研究問題

上大一時，在程式設計這堂課程中，使用了系上老師自己開發的線上自動評分系統，這套系統只能正確的判斷程式執行的結果，當時的我們也就認為程式只要結果對就好。到了大二，在軟體工程這堂課中，我們了解到了軟體程式品質的重要性，如果只在乎結果，程式中可能存在許多不必要的變數或結構，而不好的變數的宣告方式及糟糕的函式寫法都有可能產生安全性問題、影響可讀性。

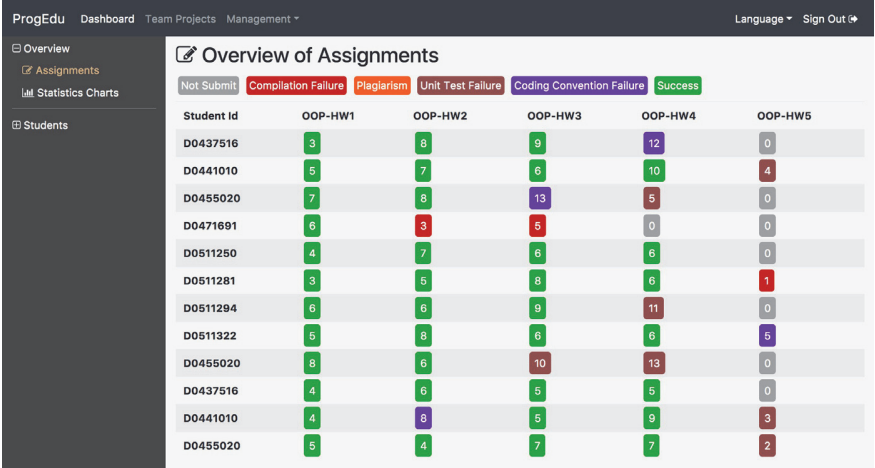
藉由這次專題的契機，我們嘗試開發出一套能判斷程式執行結果正確性，且能夠分析程式品質的教學系統。對於這套系統，我們將會開發重點放在如何檢測程式品質以及如何提升使用者程式碼品質上。檢測品質的部分，我們會使用SonarQube來檢測上面提到的三項指標：安全性、可靠性以及可維護性。提升程式碼品質的部分，在檢測完後，系統會將可改進的部分標記起來，並提供範例給使用者參考，此外，系統也會將每次結果紀錄下來，生成程式碼的品質曲線圖，以便了解學習狀況。

1. 文獻回顧與探討

在研究如何開發這套系統時，參考了現有的程式批改系統ProgEdu，我們參考了它們的系統架設方式，使用Git作為學生繳交程式碼的工具，並且加入持續整合(CI)軟體Jenkins，讓我們能即時構建並檢測學生繳交的程式碼。在品質檢測的部分，不像Progedu只檢查程式碼有無正確縮排及註解，會再加入更多的品質檢測項目，這個部份我們參考了CISQ - ISO/IEC 5055，針對程式的安全性、可靠性以及可維護性進行評量。針對這三項性質，系統能透過SonarQube檢測，並且針對特定的檢測結果能產生可直接供使用者使用的建議程式碼，這個部分參考了論文[4]。在老師出題方面，我們使用到了SCORM[5]規格一致化的概念，目標建立一個能匯入匯出系統的題庫，方便老師出題。

3-1.參考文獻回顧

1. ProgEdu：[1]是一套能自動檢查、編譯、單元測試的系統，並提供Checkstyle、Unit test等程式碼的檢驗結果，並將結果呈現在儀表板(圖一)供老師和學生檢視。



(圖一)ProgEdu將各個學生的各個作業之狀態顯示在儀表板

1. CISQ - ISO/IEC 5055:[3]根據四個關鍵因素來衡量軟體產品的內部結構： 安全性、可靠性、性能效率和可維護性。這些是決定軟體系統的可信度、

可靠性和彈性的因素。在ISO 5055之前，沒有國際標準透過分析其內部結構來衡量軟體系統的品質和完整性。這相當於只透過房屋的外觀來評估房屋，而沒有檢查其內部結構是否有木材腐爛。ISO 5055 提供了在開發中產品的事前檢查措施，以在發生問題之前識別和消除結構缺陷。

1. SCORM: SCORM(Sharable Content Object Reference Model)[5]是一套電子學習產品的技術標準。 它提供了能讓電子學習內容和學習管理系統（learning management system，簡稱LMS）協同工作的通信方法和數據模型。 它告訴程序員如何編寫代碼，以便他們構建的內容能夠與其他電子學習軟件“很好地配合”。 SCORM 是可用的最廣泛使用的電子學習標準。
2. Automatically Generating Fix Suggestions in Response to Static Code Analysis Warnings[4]：本篇論文中依據SonarQube中最常被手動更正的11個靜態代碼分析規則進行建議程式碼的撰寫，並將結果作為拉取請求 (PR) 提交到項目存儲庫中，結果顯示高達84%的請求被合併到專案中。

以下為論文中提到的11個最常被檢測出違反的規則：

B1 Strings and boxed types should be compared using equals()

B2 BigDecimal(double) should not be used

C1 String literals should not be duplicated

C2 String functions use should be optimized for single characters

C3 Strings should not be concatenated using + in a loop

C4 Parsing should be used to convert strings to primitive types

C5 Strings literals should be placed on the left-hand side when checking for equality

C6 Constructors should not be used to instantiate String,BigInteger, BigDecimal, and primitive wrapper classes

C7 "entrySet()" should be iterated when both the key and value are needed

C8 Collection.isEmpty() should be used to test for emptiness

C9 Collections.EMPTY\_LIST, EMPTY\_MAP, and EMPTY\_SETshould not be used

3-2.使用工具

1. 系統前端：預計從當今較為普遍的三種JavaScript框架Vue、React和Angular中選擇其一來建立前端使用者介面。

Vue：是一套以視圖層為基礎發展的 JavaScript 漸進式框架。Vue的目標是透過簡單的 API 提供開發者實作資料綁定與操作網頁上的元件。

React：是一個高效且具有彈性的 JavaScript 函式庫，用以建立使用者介 面。它讓你使用小巧而獨立的「component」，來建立複雜的 UI。

Angular：是一個基於TypeScript的開發平台。它將核心功能和可選功能作為一組 TypeScript 函式庫進行實現。

1. GitLab：利用Git作為版本控制系統，除了讓學生方便繳交作業，還能讓學生熟悉系統開發的流程。GitLab 是DevOps 平臺，它使組織能夠通過更快、更高效的交付軟體，同時增強安全性和合規性，最大限度地提高軟體開發的整體回報。
2. 系統後端：我們使用RESTful API來連接我們的前端和資料庫端。使用RESTful API能讓我們的API接口簡單且統一，且能讓後端API開發上好維護且有彈性。
3. 系統資料庫：我們將根據之後所決定的資料儲存方式來決定使用

RDBMS資料庫(ex:MySQL)或是NoSQL資料庫(ex:MongoDB)。MongoDB：是一個文件導向 (Document-oriented database)的資料庫管理系統的資料庫。

MySQL：是一種關聯數據庫管理系統，關聯數據庫將數據保存在不同的表中可有效增加速度並提高靈活性。

1. 程式品質檢測平台：我們使用SonarQube來協助我們對學生的程式碼

做品質檢測。SonarQube提供程式碼品質、風險檢測解決方案，目前提供了超過27種以上程式語言的開發分析。

1. 系統整合工具：我們會在系統中建置Jenkins來協助學生能快速建置

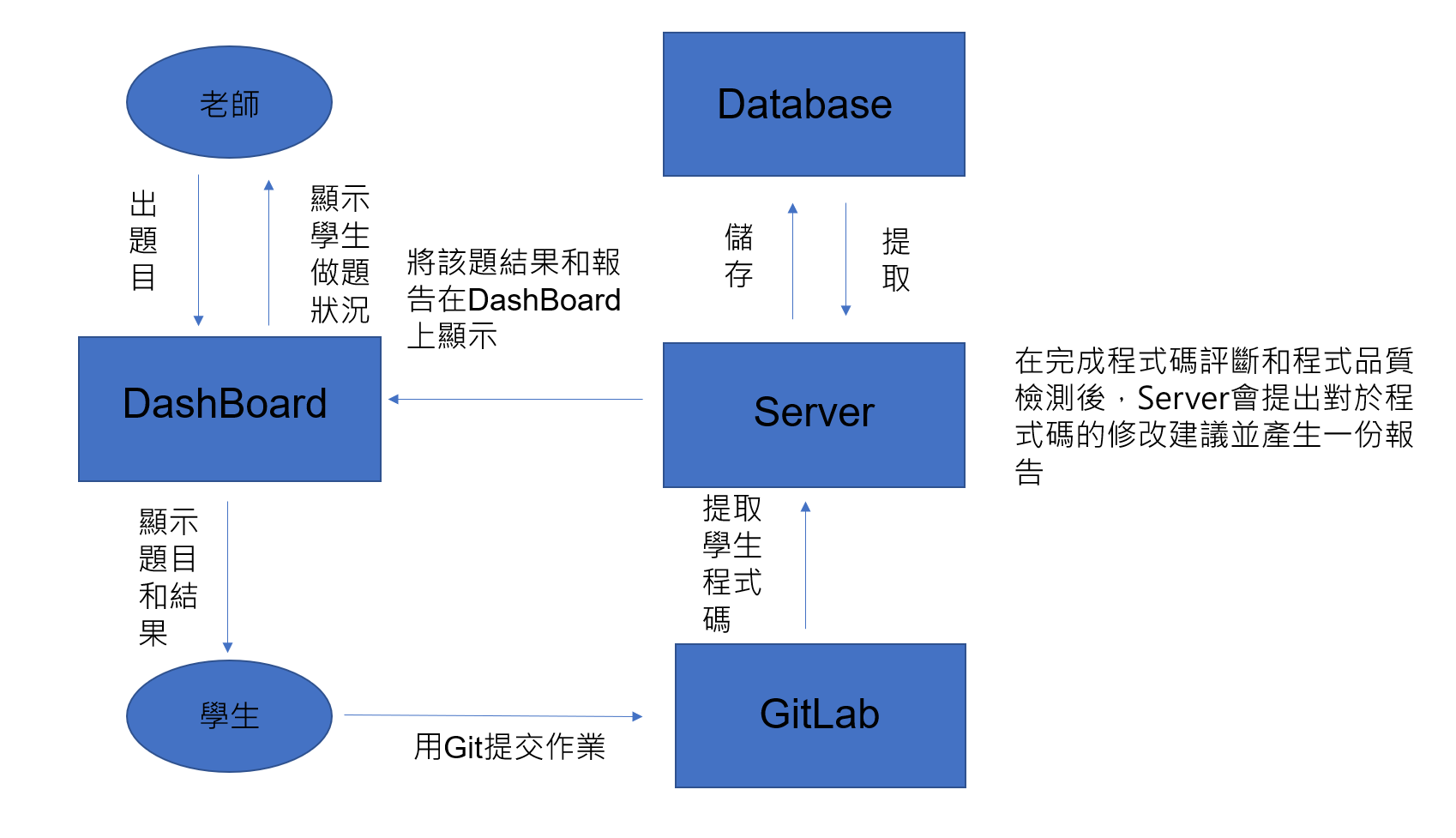
作業。Jenkins提供了軟體開發的持續整合服務，除此之外也提供相當多插件來支援不同的專案開發。

1. 系統整合容器：我們的系統將透過Docker來建置，以方便管理。

Docker是一個用於開發、發佈和運行應用程式的開放平臺。擁有輕量化、提高資源效率、更快建置等優勢。

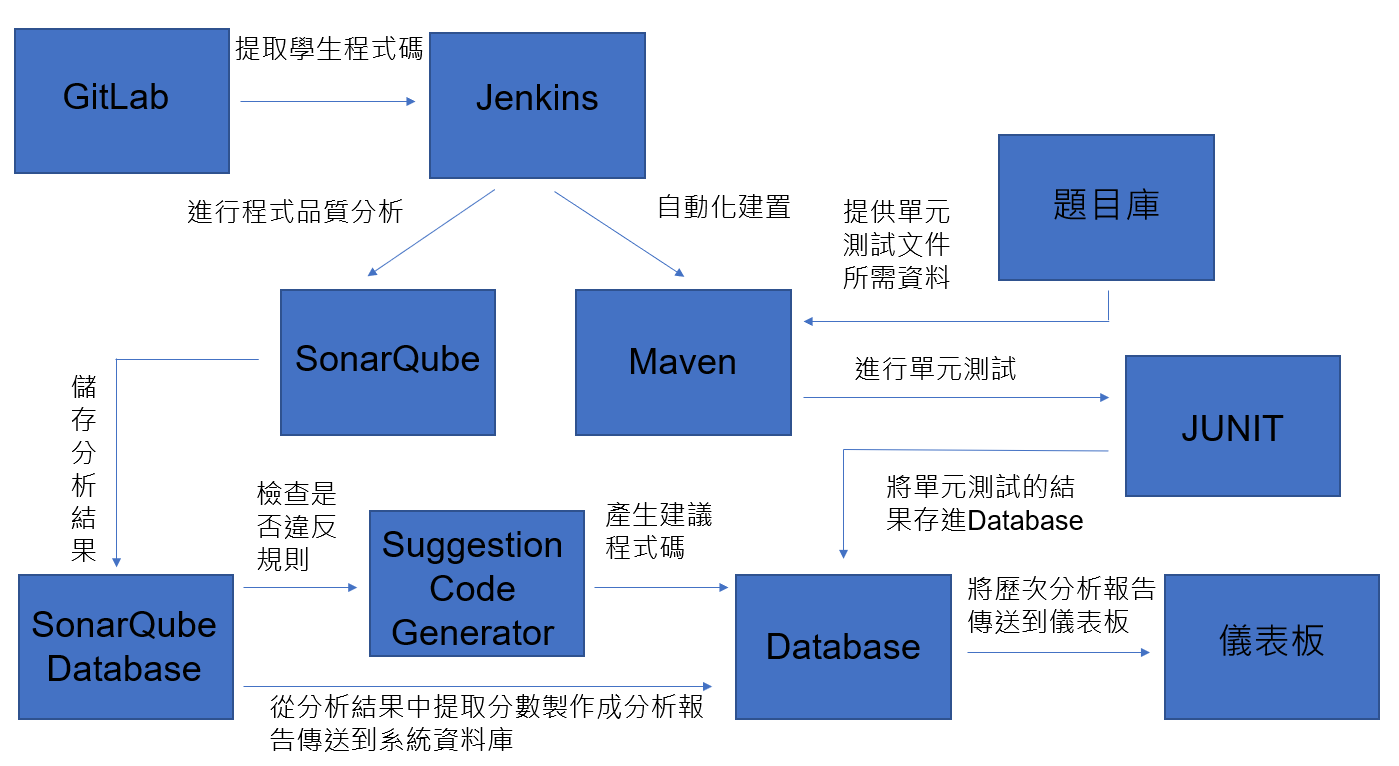
1. 研究方法及步驟

以下兩張圖分別代表我們預期系統做出來後的基本系統流程和Server內部資訊流。而為了達到我們最終的目標，我們將系統架設分成了三步驟，第一步是先架設基本的Server，第二步是在Server中加入單元測試及程式品質檢測功能，最後會是前端、題目庫以及建議程式碼產生器的架設。



(圖二)基本系統流程

目前計畫讓學生從儀表板上查看題目，完成作業後統一將撰寫的程式碼以Git提交到GitLab上，而當有檔案被上傳到GitLab時，系統會自動建置並進行單元測試，也會同時使用SonarQube檢測程式碼的品質，並將結果以PDF的格式彙整成一份報告呈現在儀表板，而老師則可以在儀表板上出題、觀看學生的做題狀況。在出題方面，會採用SCORM的概念，老師可以從網路上找題庫或是自己出題，只要符合格式，就可以出題。



(圖三)Server內部資料流

由於我們會將部件分別放入多個Docker Container中，就需要利用Docker-Compose來組合多個 container 成為一個完整服務的工具，將會用到的Docker Image都寫入docker-compose.yml中，只要下 docker-compose up 指令，就可以把所有的 Docker Container 執行起來，這樣就可以快速和方便的啟動多個 container，讓 container 的關係可以嵌入版本控制。

而Server內部的資料流程大概如下，首先，當學生將作業繳交後，Jenkins偵測到版本變更，Maven會做自動建置，我們會利用maven的test階段的maven- surefire -plugin插件來調用Junit做單元測試，SonarQube則進行程式的品質分析並產生報告。Server會利用SonarQube的web API中的Get(內容位於api/measures/ component中)，取得報告中所需的內容(包含各品質的評估等級等)，這些內容會再利用Restful API中的Post，將需要的資料存入系統的資料庫中。前端要生成報告時，會使用Restful API中的Get來取得先前儲存的資料。

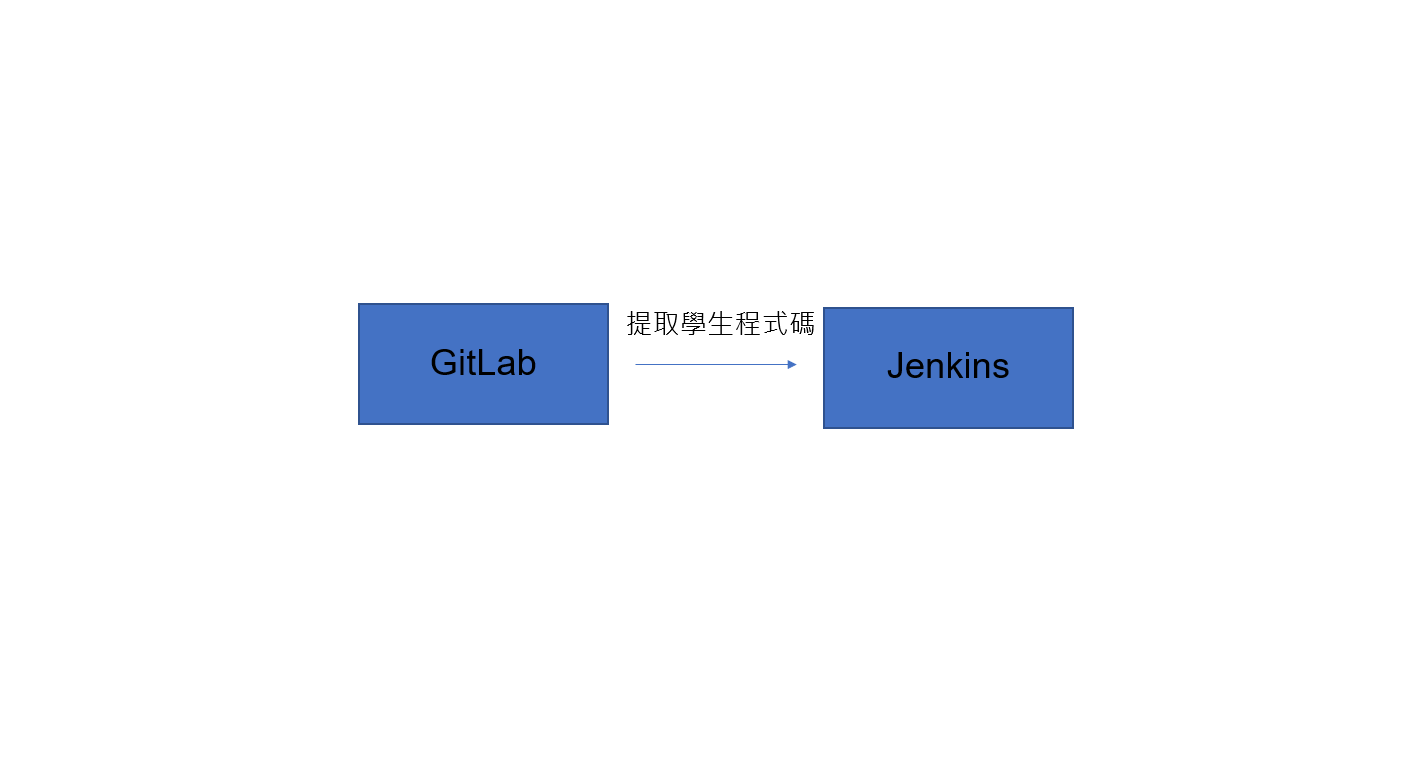
本次實驗方法分為三個階段，分別為：

階段一：基本Server建構(GitLab,Jenkins)

階段二：加入單元測試、程式品質檢測功能(SonarQube)

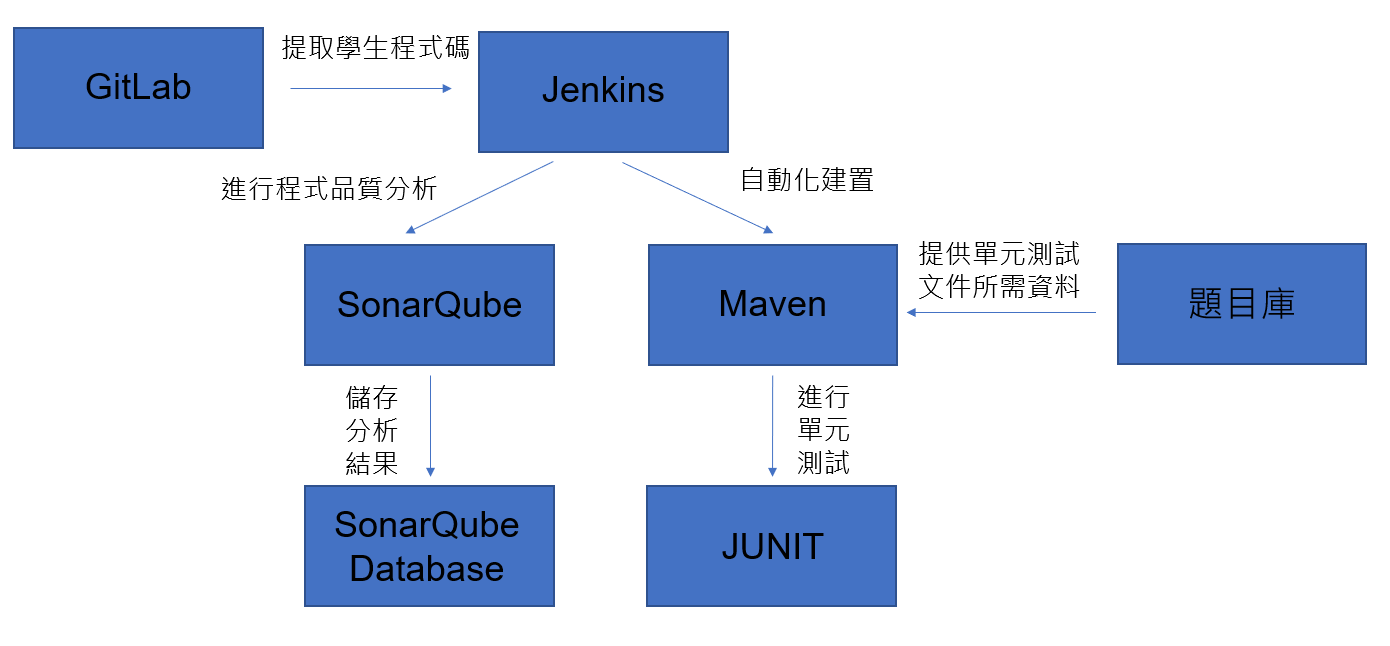
階段三：加入前端部分、資料庫、建議程式碼產生器

針對階段一的部分，我們會先建構最基礎的Server，包含GitLab和Jenkins的架設，其中GitLab會作為每個學生繳交作業的工具。Jenkins的部分，我們會在Jenkins安裝GitLab的plugin，然後在GitLab拿到SSH的金鑰，再回到Jenkins配置token，最後設定當有程式碼提交時，就會觸發Jenkins的部署操作。如此每當GitLab有新的檔案上傳時，Jenkins便會自動建置專案並對程式碼進行單元測試及品質測試。



(圖四)基本Server架構

在階段二，我們會將SonarQube加入系統中，架設SonarQube後與Jenkins連接，也會架設屬於SonarQube的資料庫，SonarQube會將程式碼掃描後的結果儲存到資料庫，我們會從掃描結果中提取安全性、可靠性和可維護性這三項指標的分數，還有SonarQube檢測出來較嚴重的項目(Blocker issues)。而單元測試則是透過Maven default生命週期的test階段綁定了maven-surefire-plugin插件，該插件可以調用Junit3、Junit4、TestNG等Java流行測試框架完成單元測試。我們會將題目庫中老師選取的題目的output抓取出來，做成單元測試的文件。



(圖五)加入單元測試和品質檢測的Server架構

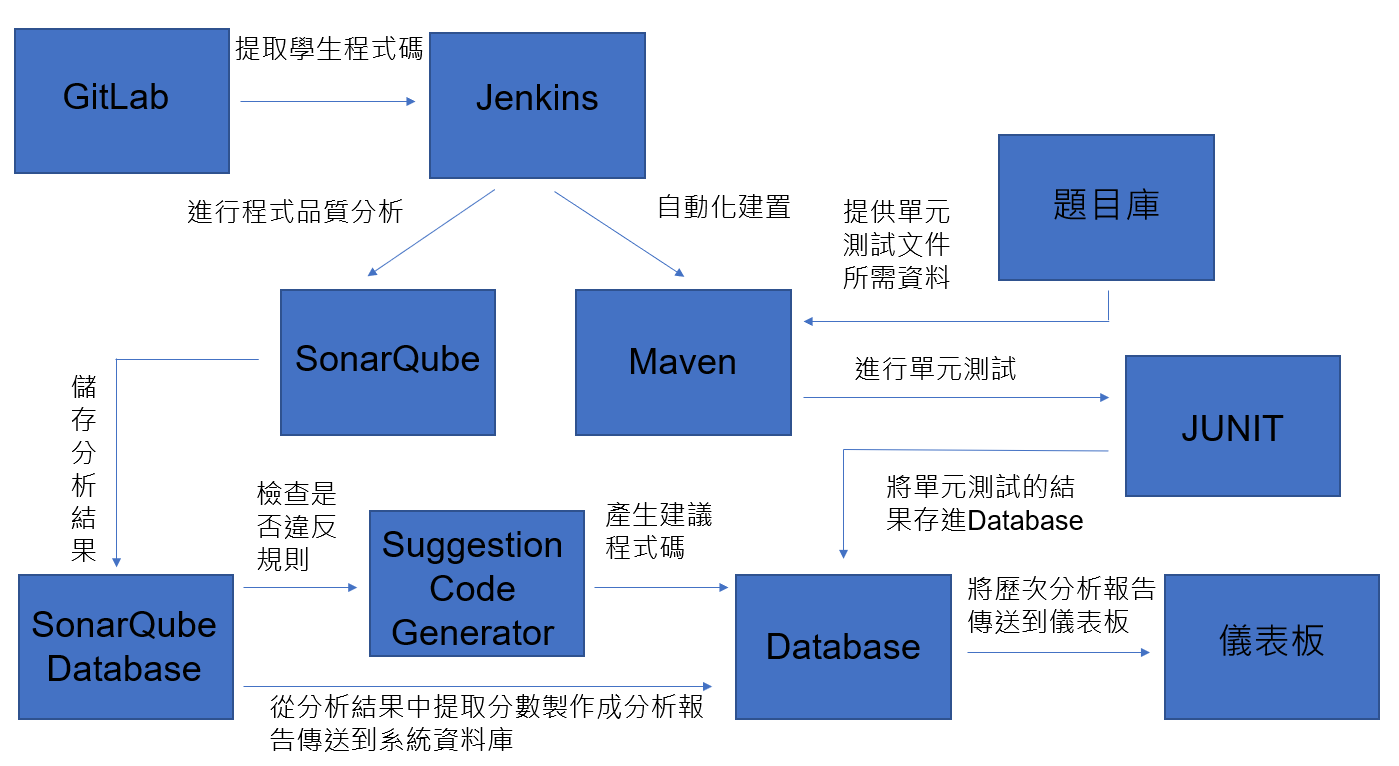
第三階段，在我們將Server架設好後，我們會開始著手處理前端顯示內容、題目庫及建議程式碼產生器。

前端的部分我們會將重點放在儀表板上，儀表板會顯示該堂課老師出了哪些作業、各個作業所有學生的繳交狀況，點進該作業自己的名字能看到自己每次作業建置的狀況和報告。

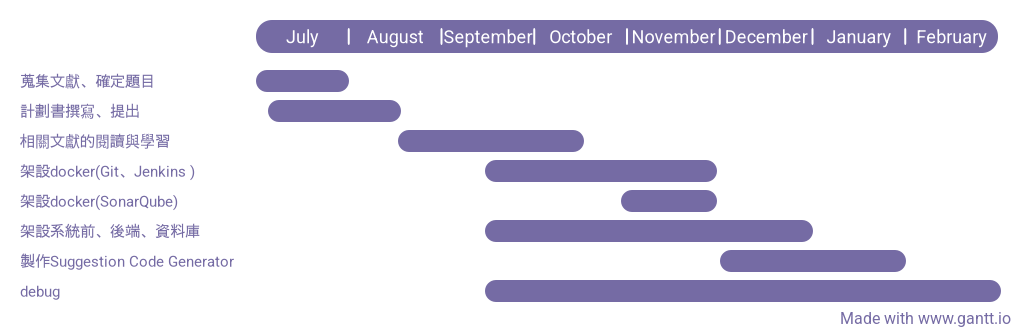
報告的部分，系統會從SonarQube Database記錄每次學生作業繳交的情況，包含每次繳交產生的bug數目，vulnerability數目，及code smell的數目，Server會將這些項目的得分儲存到資料庫，利用資料庫中各項的歷次分數來繪製學習曲線圖，藉由這個曲線來有效的提示使用者程式碼的修改趨勢，以便了解各項指標的學習狀況。

建議程式碼產生器的部分，則會參考論文[4]的實作方式，論文中提供了SonarQube中最常被手動更正的11個靜態代碼分析規則，我們會依據這些規則，製作Suggestion Code Generator，內含代碼轉換模板(code-transformation templates)，能套用到使用者的程式碼中，產生建議程式碼。而在系統確實產生建議程式碼前會先針對SonarQube檢測產生的結果進行檢查，若確實違反了其中某項規則，才會套用模板。

最後系統會將本次建置的分數、從第一次繳交到最近一次繳交的曲線圖和對於程式碼的修改建議彙整成一份報告(PDF或HTML的格式)傳送至儀表板，讓學生了解作業的作答情況。



(圖六)最終完整的Server架構

以下是我們預計的計畫甘特圖

(圖四)預計進度-甘特圖

1. 預期結果

預期功能:

(1) 提供單元測試以及品質檢測。

(2) 根據使用者程式碼錯誤提供對應的建議。

(3) 老師在出題時可以自由選擇自己出題或是從題庫出題。

(4) 提供品質曲線圖以便了解進步狀況。

(5) 報告結果整合到儀表板上。

1. 參考文獻

[1]H. -M. Chen, W. -H. Chen, N. -L. Hsueh, C. -C. Lee and C. -H.

Li, "ProgEdu - an automatic assessment platform for programming

courses," 2017 International Conference on Applied System

Innovation (ICASI), 2017, pp. 173-176, doi:

10.1109/ICASI.2017.7988376

[2]Pulasthi Perera, Roshali Silva, Indika Perera，"Improve Software

Quality through Practicing DevOps"2017 International Conference

on Advances in ICT for Emerging Regions (ICTer): 013 - 018

[3]Automated quality characteristic measures. Accessed:

2022-01-28. url: <https://it-cisq.org/standards/>

[automated-quality-characteristic-measures/.](https://it-cisq.org/standards/)

[4]Diego Marcilio,Carlo Alberto Furia,Rodrigo Bonifacio,Gustavo Pinto，"Automatically Generating Fix Suggestions in Response to Static Code Analysis Warnings"2019 DOI:10.1109/SCAM.2019.00013

[5]SCORM,url:[SCORM.com HomePage: What is SCORM and How it Works](https://scorm.com/)

(七)需要指導教授指導內容

(1)研究計畫相關方向，內容提示。

(2)引導使用研究計畫相關之文獻。

(3)成果報告撰寫。

表C802

**三、耗材、物品、圖書及雜項費用：**

1. 凡執行研究計畫所需之耗材、物品、圖書及雜項費用，均可填入本表內。
2. 說明欄請就該項目之規格、用途等相關資料詳細填寫，以利審查。
3. 依研究計畫實際需求擇優補助，每一計畫最高以補助新臺幣20,000元為限。

金額單位：新臺幣元

| 項 目 名 稱 | 說明 | 單位 | 數量 | 單價 | | 金額 | 備註 |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | |  |  |
| 合　　　　　　　　計 | | | | |  | |  |

表C803

大專學生研究計畫指導教授初評意見表

一、學生潛力評估：

二、對學生所提研究計畫內容之評述：

三、指導方式：

四、本人同意指導學生瞭解並遵照學術倫理規範；本計畫無違反學術倫理。

指導教授簽名：

年 月 日表C804