

UNDERGRADUATE PROJECT PROPOSAL

Project Title:	Web design platform based on low-code visual components
Surname:	Ruitong
First Name:	Xiao
Student Number:	201918010106
Supervisor Name:	Javier Jia
Module Code:	CHC 6096
Module Name:	Project
Date Submitted:	2022/11/15

Table of Contents

1	Introduction	3
1.1	Background	3
1.2	Aim	4
1.3	Objectives.....	4
1.4	Product Overview	4
1.4.1	Scope	4
2	Background Review	4
2.1	Summary of existing approaches.....	4
3	Methodology	6
3.1	Approach.....	6
3.1.1	Software development methodology	6
3.1.2	Requirements gathering methods	7
3.1.3	Test process	7
3.2	Technology.....	8
3.3	Version management plan.....	8
4	Project Management.....	8
4.1	Activities	8
4.2	Schedule	9
4.3	Data management plan	10
4.4	Deliverables.....	10
5	References	11

1 Introduction

1.1 Background

The rapid development of applications and the constant iteration of applications has become a new trend. Based on statistics, the average cost of a application development project ranges from US\$434,000 to US\$2,322,000, whereas 52.7% of projects will cost 89% more than predicted and only 16.2% of projects will ultimately complete on time, with even 31.1% of projects being cancelled before completion [1]. Additionally, due to a 38% reduction of job growth in 2021, while market demand in the IT field is constantly being updated, in the next few years, thousands of job vacancies might be created, recruiting development engineers is becoming ever more difficult, companies are facing the challenge of not being able to meet the market requirements for new applications in time and having to find faster and cheaper ways to meet their application requirements [2]. In response, the Low Code Development Platforms (LCDPs) has been produced and attempts to provide a quick development method.

With a growing trend of Low Code Development Platforms (LCDPs) which provide a viable solution to the shortage of highly skilled professional developers by allowing "people with no education in programming", i.e. citizen developers [3], to participate in the application development process through higher-level abstractions. According to a concept proposed by Forrester in 2014, LCDPs are being used by citizen developers or non-programmers to build applications that meet their needs based on the drag and drop of components, visual modelling and automatic code generation [4]. Furthermore, LCDPs are conducive to improving the efficiency of application development and significantly reducing the cost of traditional application development projects, addressing the high cost and inefficiency of enterprise application development [5]. Research shows that low-code development is also the most common solution to the skills shortage of development professionals, and that low-code development platforms not only reduce development costs and time, but also produce high-quality products [6].

By 2023, the market for LCDPs is predicted to grow sharply, with more than half of medium to large enterprises will adopt LCDPs as one of their top strategic application platforms [7].

1.2 Aim

The purpose of this paper is to provide a low-code development platform based on a visual environment and drag-and-drop components to enable citizen developers to design websites design websites quickly and cost-effectively, thereby addressing the high cost and inefficiency of traditional application development as well as the shortage of developers.

1.3 Objectives

- a. Explore the project background, functional priorities and requirements analysis.
- b. Decide on the project development environment and resources.
- c. Design the user interface and visualization components.
- d. Implement the front and back-end development of the project.
- e. Website testing and maintenance

1.4 Product Overview

1.4.1 Scope

The project provides a fast and low-cost, low-threshold application development service that eliminates the programming process and facilitates developers with an advanced graphical user interface and drag-and-drop component design that enhances the user interaction experience, in order to enable everyone to participate in the website development process.

The platform accelerates the development of applications by reducing the programming process, the main target audience is citizen developers. Unlike professional developers who build solutions using traditional programming, citizen developers are essentially business users with little to no background in application development, and the low-code platform allows with no knowledge of coding and provides them a code-free development experience.

2 Background Review

2.1 Summary of existing approaches

More and more enterprises have been turning to low-code development platforms in recent years. Below is a table of the existing platforms and their features in comparison.

Products	Strengths	Weaknesses
Microsoft Power Apps	High abstraction level: Provides drag and drop tools with a high abstraction level.	Constraint: Available only under a traditional PaaS model, which is not a match for customers who must deploy applications on-premises or with other cloud providers.
	API and integration services: Has a rich set of APIs and OData endpoints (execute full create, read, update and delete (CRUD) operations on data in order to support integration with third-party iPaaS.), has sufficient data connectors.	Sales execution and pricing: Complex pricing models and entitlement rules, and some features require extra payments.
Mendix Platform	Functionality: Robust functionality for integration, event processing, workflow and development support using AI	Constraint: Content management only includes the basics of file management and storage
	Innovation: Provides highly customizable chatbots, mobile apps and IoT applications.	
	Overall viability:	Hysteresis:

Oracle Application Express (APEX)	Has a huge number of audiences	Lags behind in modern features.
	Extendibility: Extended with JavaScript, Java and SQL, APEX applications are available to access all the advanced features of the Oracle database platform.	External dependencies: A heavy reliance on PL/SQL, supplemented by Java and JavaScript for business logic
Studio Creatio	Data processing and validation: Providing data processing models and validation.	Lack of teamwork functionality: Lack of multi-cloud/multi-region deployments and seamless auto-scaling of applications
	Matches customers' requirements: Allows citizen developers to create drag-and-drop case design features and their own custom machine learning models.	Innovation: Lags behind in integrating functionality and innovative application development

Table 1. Background Research and Competitive Analysis [8]

3 Methodology

3.1 Approach

3.1.1 Software development methodology

Developing projects with incremental models treats the project product as a series of incremental building blocks to be designed, coded, integrated and tested. Each building block consists of multiple interacting modules and is capable of performing specific features.

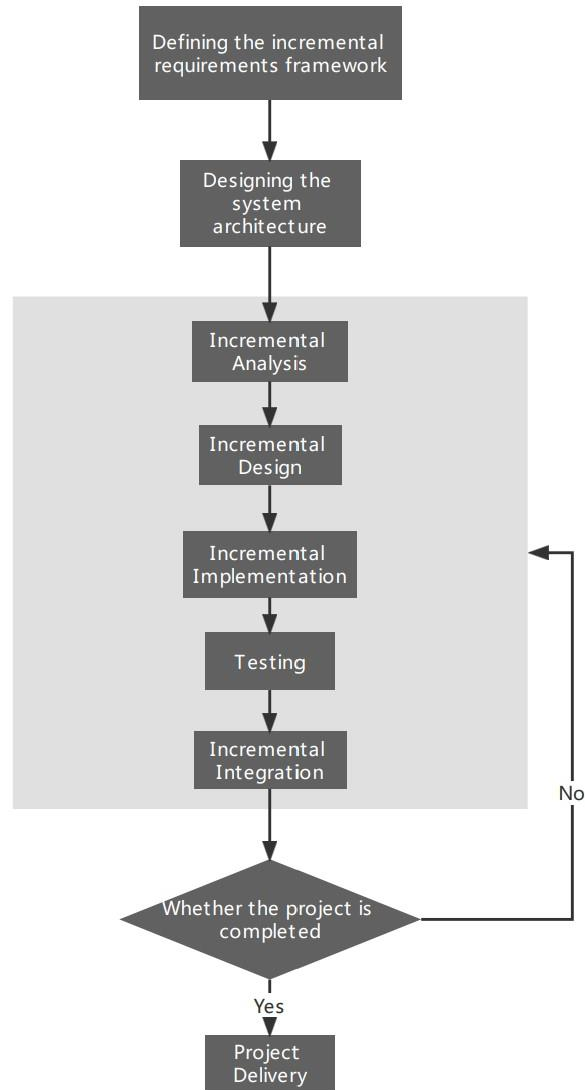


Figure 1.Incremental Development Model Flowchart

3.1.2 Requirements gathering methods

Regarding the requirements gathering method, this project will use benchmarking to compare this product with other leading products in the same field, in order to determine the objectives of this product and the implementation methods, as well as to make incremental improvements based on existing methods.

3.1.3 Test process

The functional testing process for this project:

- a. Link testing
- b. Form testing
- c. Search test
- d. Deletion test
- e. Cookies, session testing
- f. Database testing

3.2 Technology

The project relies on the cloud computing technology to integrate hardware and software resources, using Ali cloud mirror source service to build cloud servers, MySQL to achieve database management. The back-end development is integrated with Springboot and Mybatis-plus, while the front-end utilizes React to provide separate front-end and back-end development. The main tools chosen for development are IDEA and Visual Studio Code, with Navicat as the database management tool.

3.3 Version management plan

Use GitHub to manage project versions and project-related files, and update them after each development phase.

GitHub Repository: [AuroraXiao/Web-design-platform-based-on-low-code-visual-components: 201918010106 Aurora \(github.com\)](https://github.com/AuroraXiao/Web-design-platform-based-on-low-code-visual-components)

4 Project Management

4.1 Activities

The following are the activities required for each objective:

1. Explore the project background, functional priorities and requirements analysis.
 - 1.1 Conduct literature search
 - 1.2 Analyze and compare competitors
 - 1.3 Defining the project objectives
 - 1.4 List item functions and prioritize project functions

- 1.5 Analysis of user requirements from role, scenario and time
 - 2. Decide on the project development environment and resources.
 - 2.1 Explore the existing front and back-end technologies
 - 2.2 Select the most suitable development technologies
 - 2.3 Select the appropriate development tools
 - 3. Design the user interface and visualization components.
 - 3.1 Complete prototype diagrams, information architecture diagrams, flowcharts
 - 3.2 Complete UI diagrams based on prototype diagrams
 - 3.3 Collection of common front-end components and icon libraries
 - 4. Implement the front and back-end development of the project
 - 4.1 Front-end interface development from UI diagrams
 - 4.2 Implementing back-end interface development
 - 4.3 Create and connect to database
 - 5. Website testing and maintenance
 - 5.1 Defining the test environment and test plan
 - 5.2 Test and maintain website functionality until delivery
- 4.2 Schedule

The following is a schedule of the project development process, which shows the entire project phase.

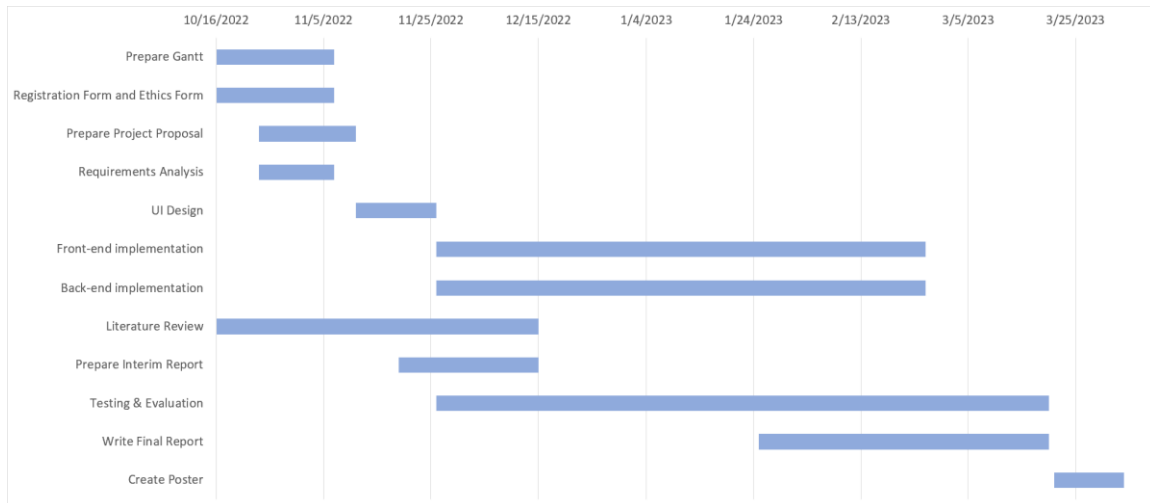


Table 2. Gantt Diagram

4.3 Data management plan

The project's reports, weekly report sheets are stored in the FeiShu cloud, allowing students and supervisors to update the project content and provide feedback.

Back up and upload all relevant literature to Mendeley for management and citation of the literature.

Weekly Report Sheets Link: [\(201918010106 Aurora\) Weekly Report Sheets](#)

Project Report Link: [\(201918010106 Aurora\) Project Report](#)

4.4 Deliverables

- a) Project Proposal
- b) Ethics Form E1
- c) Progress Report
- d) Final Report
- e) Weekly Report Sheet
- f) Poster Presentation
- g) Project Code

5 References

- [1] M. Fryling, 'Low Code App Development', *J. Comput. Sci. Coll.*, vol. 34, no. 6, p. 119, Apr. 2019.
- [2] T. Breaux and J. Moritz, 'The 2021 software developer shortage is coming', *Commun ACM*, vol. 64, no. 7, pp. 39–41, Jul. 2021, doi: 10.1145/3440753.
- [3] M. Oltrogge *et al.*, 'The Rise of the Citizen Developer: Assessing the Security Impact of Online App Generators', in *Proceedings - IEEE Symposium on Security and Privacy*, Jul. 2018, vol. 2018-May, pp. 634–647. doi: 10.1109/SP.2018.00005.
- [4] V. Phalake, S. Joshi, K. Rade, and V. Phalke, 'Modernized Application Development Using Optimized Low Code Platform', in *2022 2nd Asian Conference on Innovation in Technology (ASIANCON)*, Aug. 2022, pp. 1–4. doi: 10.1109/ASIANCON55314.2022.9908726.
- [5] A. C. Bock and U. Frank, 'In Search of the Essence of Low-Code: An Exploratory Study of Seven Development Platforms', in *Companion Proceedings - 24th International Conference on Model-Driven Engineering Languages and Systems, MODELS-C 2021*, 2021, pp. 57–66. doi: 10.1109/MODELS-C53483.2021.00016.
- [6] S. Rafi, M. A. Akbar, M. Sánchez-Gordón, and R. Colomo-Palacios, 'DevOps Practitioners' Perceptions of the Low-code Trend', Sep. 2022, pp. 301–306. doi: 10.1145/3544902.3546635.
- [7] P. Vincent *et al.*, 'Licensed for Distribution Magic Quadrant for Enterprise Low-Code Application Platforms', 2020. [Online]. Available: <https://www.gartner.com/doc/reprints?id=1-24BBDEDZ&ct=201005&st=sb>
- [8] J. Bratincevic and R. Koplowitz, 'The Forrester Wave™: Low-Code Development Platforms for Professional Developers, Q2 2021 The 14 Providers That Matter Most And How They Stack Up Why Read This Report', 2021.