Artifact Agreement

Project Title: Simple Accounting Package
Course Name: IFB399
Instructor: Chris Koren
Team Members: Lachlan Thompson, Harrison Kelly, Lenny Wang, Anthony Chen

Purpose

This agreement outlines the terms under which software artifacts—including but not limited to source code, documentation, and design documents—will be created, shared, and managed for the **Simple Accounting Package** project.

Scope

This agreement applies to all artifacts created for the **Simple Accounting Package** during the period of 24/07/23 to 23/10/23.

Ownership and Intellectual Property

All artifacts produced under this agreement are collectively owned by the team members listed above and are subject to the intellectual property guidelines set forth by Queensland University of Technology.

Deliverables

The following artifacts will be created and delivered:

Source Code
Design Documents
User Manuals
Test Plans

The following functions will be created and delivered upon:

ID	Category	Feature	Description	Must/Should/Could/May Have	Will/May/Won't Deliver
1	Invoicing	Invoicing History	Invoicing will have features such as invoicing history, which can be accessed at a later date.	Should have	Will deliver
2	Invoicing	Invoicing History	Invoices that have been sent will be moved to a invoicing history table	Must have	Will deliver
3	Invoicing	Invoicing Generation	This will allow users to generate an invoice by inputting data into a number of fields. The invoice will then be generated for them.	Must have	Will deliver
4	Invoicing	Invoicing Generation	Users can then view individual invoices that will dynamically display the data that was stored when generating an invoice.	Must have	Will deliver
5	Invoicing	Invoicing Workflow	Users can download invoices as a pdf on their local file system.	Must have	Will deliver
6	Invoicing	Invoicing Workflow	Users can send the invoices via email to their customers.	Should have	May deliver
7	Invoicing	Invoicing Generation	Calculate GST per invoice	Must have	Will deliver
8	Invoicing	Notifications	Users will receive a notification when a new invoice is received	Should have	May deliver
9	Quoting	Quote History	Quoting will have features such as invoicing history, which can be accessed at a later date.	Should have	Will deliver

10	Quoting	Quote Generation	Users can generate an invoice by inputting data into a number of fields. The invoice will then be generated for them.	Must have	Will deliver
11	Quoting	Quoting Generation	Users can then view individual quotes that will dynamically display the data that was stored when generating a quote.	Must have	Will deliver
12	Quoting	Quoting Workflow	Users can download quotes as a PDF on their local file system.	Must have	Will deliver
13	Quoting	Quoting Workflow	Users can send the quotes via email to their customers.	Should have	May deliver
14	Quoting	Quoting Generation	Calculate GST per quote	Must have	Will deliver
15	Quoting	Notifications	Users will receive a notification when a new quote has been accepted	Should have	May deliver
16	Financial Management	Payroll	Users can input their hours and job code into the system to calculate total weekly/monthly hours worked	Should have	May deliver
17	Financial Management	Payroll	Users will be able to calculate their time in lieu	Should have	May deliver
18	Financial Management	Payroll	Users will be able to input the work hours based on their employment type	Should have	May deliver
19	Financial Management	Payroll	Users will have their overtime calculated based on their employment type. Different rules to apply to casual, part-time or full-time	Should have	May deliver
20	Financial Management	Payroll	Users will have the option to charge a specific job code	Could have	Won't deliver

			for hours worked on different projects		
21	Financial Management	Reporting	Users can generate financial reports for ASX announcements	Could have	Won't deliver
22	Financial Management	Payroll	Users can generate financial reports on profit and loss	Could have	Won't deliver
23	Financial Management	Payroll	Users can generate tenement financial reports inline with legislative requirements	Could have	Won't deliver

Timeline

All artifacts should be completed and submitted by 23/10/23.

Quality Standards

Team Balanced will provide a standard of quality to this project that will be acceptable by the standards of both parties (OreFox and Team Balanced). Specifically, the quality standards will consist of many areas such as:

- Code Quality
  - All code related to the accounting package will be clearly written.Variable and function names related to invoicing, quoting and payroll management will be descriptive and meaningful.
  - All code follows a consistent style. Team Balanced will follow a specific coding standard.
  - The code will be well-commented as to show what each section does and what purpose it has. This is to ensure any future work done on the code is simple and easier to implement.
- Documentation Quality
  - Ensure that the logbook is updated regularly, capturing key decisions, changes, and progress made throughout the project.Entries in the logbook should be clear and concise, providing enough detail for any team member to understand the context and rationale behind decisions.
  - The report should follow a structured format, including sections like introduction, objectives, methodology, results, and conclusion. All claims or statements in the report should be backed by evidence, such as data, screenshots.
  - User manuals should provide step-by-step guidance on how to use the software, with clear screenshots and examples.

- All aspects of the project, from design to implementation, should be documented. Any gaps or missing information should be identified and addressed.
- Testing Quality
  - Ensure that a variety of tests are conducted, covering different functionalities and scenarios within the software.
  - Test scenarios that are less common but could occur, ensuring the software handles all possible situations gracefully.
  - Regular functionalities, especially core features like invoicing or financial management, should be thoroughly tested to ensure they work as intended.
- Useability
  - UI Design
    - The user interface should maintain a consistent design across all modules. This includes consistent button placements, color, style and typography.
    - Users should receive clear feedback after performing actions, such as successfully creating an invoice or encountering an error.
  - Efficiency
    - Users should be able to easily change between different sections, with a clear and intuitive menu.
- Security
  - Access is granted only to logged in users as a means to protect sensitive data from public access.

The quality standards outlined above will be applied to provide a usable and working web application for OreFox.

Use and Sharing

Team members have the right to use the artifacts for educational purposes and to include them in academic portfolios. Any use outside of this context must be mutually agreed upon in writing by all team members.

Confidentiality

All discussions and documents related to this project should remain confidential unless agreed otherwise by all team members.

Amendments

Changes to this agreement must be approved by all team members and the course instructor.

Agreed Upon and Accepted:

Lachlan Thompson n11023015
10/09/23

Yi Wang n11401184
13/09/23

Harrison Kelly n9986189
13/09/23

Yiyu Chen n11400862
13/09/23