

**PROG6001 Managing Software Development Projects**

**ASSESSMENT 1 COVER SHEET**

For use with online submission of the report

Please complete all of the following details and then make this sheet the **first page of your report – do not send it as a separate document.**

**Group No:** \_\_\_

**Students Details: Piyush Kalyanbhai Parmar - 24541075**

**Unit Name: Managing Software Development Projects**

**Unit Code: PROG6001**

**Assignment No.: Assignment 1,**

**Assignment Title: Project Proposal Document**

**Due date: 25th March 2025**

**Date submitted: 25th March 2025**

Table of Contents

[Executive Summary 3](#_Toc193831749)

[Background 3](#_Toc193831750)

[History 3](#_Toc193831751)

[User Stories 3](#_Toc193831752)

[Testing Strategies (User Acceptance Testing) 3](#_Toc193831753)

[Goals 4](#_Toc193831754)

[Deliverables 4](#_Toc193831755)

[Timeframe 4](#_Toc193831756)

[Change Management 5](#_Toc193831757)

[Version Management 5](#_Toc193831758)

[System Building 6](#_Toc193831759)

[Release Management 6](#_Toc193831760)

[Conclusion 6](#_Toc193831761)

[References 7](#_Toc193831762)

Project Proposal: Mobile App Upgrade with PayID Scanner and DevOps Process Optimization

# Executive Summary

The mobile application of the bank will receive a PayID payments enhancement through the addition of QR code scanning features according to this proposal. The existing Osko payments process using PayID requires user involvement for manual exchange of details because it creates errors and reduces efficiency. The application's QR code scanner will simplify PayID transactions by carrying out automated information filling thereby minimizing errors. The development project involves testing at various stages with UAT among them as part of a deployment initiative to enhance payment precision and user satisfaction levels (Rahi et al.)

# Background

## History

PayID functions under the New Payments Platform (NPP) in Australia to enable users to attach bank accounts to accessible identifiers such as mobile numbers or email addresses for fast and instant Osko payments. The current approach for entering PayID information through manual entry stands as a potential cause for payment errors and delays in transactions. Data entry errors and transaction delays would be minimized through automated entry methods which introduce a QR code scanning feature ("PayID® Fast Secure Payments," 2025)

## User Stories

1. Bank customers should have the ability to use a QR code scanner for autofilling PayID payment details to reduce manual errors and speed up transactions.
2. My focus as a mobile app developer includes developing the PayID QR code scanner to work without interruption with current payment systems while sustaining both safety protocols and user-friendly features.
3. Within my compliance officer role, I must validate how the new feature upholds all financial regulations while safeguarding user data to maintain excellent bank reputation and avoid legal consequences. ("How to send and receive money with PayID - CommBank," 2025)

## Testing Strategies (User Acceptance Testing)

These UAT strategies will guarantee that the feature complies with user requirements and regulatory specifications:

* The UAT verifies how well the QR code scanner recognizes PayID details through various devices while operating in different conditions.
* The UAT strategies include testing of security features to maintain both financial compliance and protection against various threats like fraudulent QR codes and data breaches.
* The evaluation of QR code scanning functionality in the application will consist of usability testing to gather user feedback about the process's ease of use.
* Performance Testing: Evaluate the feature’s responsiveness and its impact on the app’s overall performance.

User acceptance of QR code mobile payment systems depends on how useful people perceive the system to be, how easy it is to use and how convenient it makes their transactions (Türker et al., 2022)

The design along with testing stages require these elements for achieving successful adoption.

# Goals

* The mobile application should receive a QR code scanning functionality which will enable automatic PayID detail input.
* The system should enhance the accuracy of transactions while cutting down time needed for Osko payment completion by customers.
* The new feature needs to match all necessary financial and security standards.
* The payment process must become easier to enhance client satisfaction in all aspects.

# Deliverables

* A fully operational QR code reading feature will be included inside the current mobile application.
* The user interface contains modified elements for accommodating the new scanning functionality.
* Comprehensive testing reports covering functionality, security, usability, and performance.
* The project delivers complete training resources together with support documentation for the customer support staff.

# Timeframe

|  |  |
| --- | --- |
| Phase | Duration |
| Requirement and Design | 2 Weeks |
| Development and Integration | 6 Weeks |
| Implementation | 4 Weeks |
| UAT and Deployment | 3 Weeks |

This feature helps the bank boost the accuracy combined with efficiency of Osko payments to ensure high customer satisfaction and maintain its position as a sector leader in digital banking.

To address the challenges in consistency and collaboration within the development team, the following strategies will be implemented across key areas of the development lifecycle: change management, version management, system building and release management. These strategies are intended to streamline workflows, improve communication, and minimize integration issues and delays.

# Change Management

Change management effectiveness plays an essential role to track software variations properly and efficiently implement them. I will create an organized change management procedure which contains procedural steps to process requests for change reviews and approvals. Changes will be tracked and rollbacks made possible by implementing Git as a version control tool to support the process (Sommerville, 2011). The implementation of a collaborative platform like Jira or Trello will serve to document along with track change requests and their related tasks.

The project team will conduct periodic meetings to examine ongoing modifications which keep all members consistent with project objectives. Alongside this measure developers will avoid work conflicts which occur when teams separate themselves leading to improved interteam transparency with better communication. A systematic versioning policy will be put into action to maintain orderly integration of modifications into the present codebase without disrupting any previous functionalities (Cohn, 2010).

# Version Management

The system will properly handle and integrate various software versions thanks to version management protocols. Lack of standardized version control methods between developers causes difficulties when integrating their work because they strive with different code versions concurrently. I will establish Git branching strategies including feature branches together with develop branches and release branches to minimize this problem. Developers can use feature branches for separate coding activities whereas the develop branch functions as the official integration area for test-ready features. Release branches serve to retain stable production code that stays separate from development processes. Every software version will feature version tagging while releases will get detailed documentation which facilitates easier progress tracking along with bug detection between versions (Atkins et al., 2002).

# System Building

System building transforms software programs into ready-to-deploy formats. Delays occur because the system lacks standardized building and integration processes between its components. Introducing Jenkins or Travis CI automated build systems will play a key role in speeding up system building by running automatic compilations of software anytime code in the database receives adjustments. Identification of errors takes place during early development stages through this system which allows developers to receive immediate feedback. The automated systems establish continuous integration (CI) testing features that run tests on the latest version on a continuous basis. Early conflict detection happens through this method which allows for quick resolution and more uniform software build results. (Fowler, 2006)

# Release Management

Software deployment operations into production are managed through planning and scheduling as well as controlled deployment activities. Badly orchestrated release deployment procedures lead to deployment failures and irregular releases in the system. Through a planned and automatic implementation I will put into place an organized release management approach which defines structured release schedules along with automated pipeline processes andtests before every deployment step.

The company will build a staging environment for production-level tests which will run complete UAT tests on each release. Release dates will be displayed on a planned calendar system for stakeholders to organize their product delivery schedules (Lehman & Ramil, 2003).

# Conclusion

The development team can address integration problems to decrease delays through the implementation of structured change management and version control practices together with automated system building and systematic release management. These tactical approaches build superior team collaboration while cutting down disagreements to deliver an optimized deployment procedure for our software development process.

# References

Atkins, D. L., Ball, T., Graves, T. L., & Mockus, A. (2002). Using version control data to evaluate the impact of software tools: A case study of the version editor. *IEEE Transactions on Software Engineering*, *28*(7), 625-637.

Cohn, M. (2010). *Succeeding with agile: software development using Scrum*. Pearson Education.

Fowler, M. (2006). Continuous integration. martinfowler. com. *URL:* [*https://martinfow*](https://martinfow) *ler. com/articles/continuousIntegration. html (visited on 11/29/2023)*.

How to send and receive money with PayID - CommBank. (2025). *Commbank.com.au*. <https://www.commbank.com.au/digital-banking/pay-id/send-and-receive-money.html>

Lehman, M. M., & Ramil, J. F. (2003). Software evolution—Background, theory, practice. *Information Processing Letters*, *88*(1-2), 33-44.

PayID® Fast Secure Payments. (2025). *Westpac.com.au*. <https://www.westpac.com.au/personal-banking/online-banking/making-the-most/payid/>

Rahi, S., Zaheer, M., Alghizzawi, M., & Ngah, A. H. Investigating the role of innovativeness, technology acceptance model and theory of planned behaviour towards adoption of QR mobile payment system.

Sommerville, I. (2011). Software engineering (ed.). *America: Pearson Education Inc*.

Türker, C., Altay, B. C., & Okumuş, A. (2022). Understanding user acceptance of QR code mobile payment systems in Turkey: An extended TAM. *Technological Forecasting and Social Change*, *184*, 121968. <https://doi.org/https://doi.org/10.1016/j.techfore.2022.121968>