

# SonamR\_A1\_ProjectProposal

*by* Sonam Rani

---

**Submission date:** 24-Mar-2025 09:24PM (UTC+1100)

**Submission ID:** 2623633129

**File name:** SonamR\_A1\_ProjectProposal.docx (48.38K)

**Word count:** 1286

**Character count:** 7896



## **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:** 03

**Students Details:**

Name: Sonam Rani ID: 24602309

**Unit Name:** Managing Software Development Projects

**Unit Code:** PROG6001

**Assignment No.:** Assignment 1,

**Assignment Title:** Project Proposal Document

**Due date:** 24 March 25

**Date submitted:** 24 March 25

**PROJECT PROPOSAL**  
**(QR Scanner Integration)**

**PART A:**

**1. Executive Summary:** Our mobile banking application needs to receive an upgraded version of QR code scanning for its existing PayID payment solution. Users currently complete manual PayID details sharing, for which errors occur to reduce the quality of the customer experience. PayID QR code scanner integration will establish an easy-to-use payment system that helps Australia progress with NPP real-time payments according to (Pillai, 2023).

**2. Background:**

**History:** With PayID, users can perform more convenient transactions using their phone, email addresses, or ABN as alternative payment methods to traditional complex bank account information. Users dealing with manual PayID identifier sharing encounter problems because they must handle the identification details manually. Australia uses the New Payments Platform (NPP) as its real-time payment system to provide efficient secure seamless financial transactions between banks instantly. The solution enables users to accomplish rapid secure transactions between banks utilizing the NPP "Osco" layer service.

**User Stories:**

Customer: Instantly pay a friend/Seller/Merchant using a QR Code

I will use my banking app to scan my friend's PayID QR code, immediately populating payment entry fields. After using the automatic details entry system, I can send payments without manually inputting PayID information. The following are the steps:

- Select "Pay with QR code".
- People with PayID QR codes can access their details by scanning them on their mobile phones.
- PayID information automatically appears, showing the valid payee identification.
- Input the amount and pay.

- The system confirms the successful completion of transactions by sending immediate Osko notification feedback.

Seller/ Merchant: Receive payments via QR code

I will create PayID QR codes that link to my business account while placing them in each store location for customers to use as a quick payment method. The following are the steps:

- Users can produce their QR codes through the banking application.
- The PayID details associated with the bank account are embedded within the QR code.
- The system allows merchants to obtain and show the printed QR codes.
- The merchant's payment details will automatically appear when customers scan the QR code.
- The merchant will receive instant alerts for complete transactions.

**3. Testing Strategies (User Acceptance Testing - UAT):** Users will perform tests on the PayID scanner during practical application. Users need to scan multiple QR codes containing different PayID details to validate that the software automatically extracts and handles data entry. After the scan, users will test the payment process function to verify proper transaction completion. The UAT process will find usability issues or system defects before formal release to guarantee the feature correctly serves customer requirements.

**4. Goals:** The bank plans to introduce PayID scanning to their mobile application because they want to make payments easier for users. Scannable QR codes through this enhancement simplify the manual task of entering PayID information. The application's payment flow system is set to maximize efficiency by minimizing human mistakes during manual data entry thus creating a more user-friendly experience for payments.

**5. Deliverables:** The project will integrate complete PayID scanner functionality through the mobile application to enable users for fast payments by scanning QR codes. The organization will track all documentation from start to finish of testing while producing extensive results from User Acceptance Testing (UAT) sessions.

User instructions paired with complete guidance documentation will support the feature implementation because they lead to a seamless transition towards user acceptance.

- 6. Timeframe:** The project spans twelve weeks through which developers execute planning and design activities from week 1 to week 4. The development period along with internal testing phases span from weeks 5 through 8. The User Acceptance Testing will run from weeks 9 to 10 to merge user feedback into product improvements. The deployment process of the project together with final adjustments occurs in weeks 11 and 12 before the project shifts to operational use.

**Part B:**

**Proposal for Process Improvement of Software Deployment Consistency**

Product deliveries have always been a persistent challenge for our development team. Team structures imposed by developers hinder both internal communication among teams and external collaboration which leads to software problems along with integration disturbances that ultimately delay project delivery. Our team proposes various developments to improve change management, version management systems, system construction, release management, and approaches to solve existing issues.

Software variations lead to modifications of system operations along with changes in control requirements before and after implementing and conducting audits on changed systems. The transition to Agile methods enables programmers to work with greater cooperation while improving open communication between developers since they operated independently in the past, resulting in fewer integration issues (Cockburn & Highsmith, 2001). Systems implementing tools through Jira or Azure DevOps need formal change assessment followed by authorization to become more accountable but also experience increased ambiguity. Through utilizing centralized tools for documentation, such as Confluence and Jira, we achieve better visibility apart from better documentation, which supports independent decision-making and subsequent integration (Ambler & Lines, 2012).

A solution for version control tackles the problems caused by independent work and constant integration issues. A centralized version control system known as Git delivers enhanced shared software development history, branching and merging features within GitFlow. Through the pull request review process, the team ensures software quality and knowledge sharing and detects errors in advance to produce more robust finalized software.

The development process keeps software components well-integrated during system construction while reducing production cycles. The CI framework utilizes Jenkins or GitLab CI/CD tools to perform automated code compilation testing and verification until integration bugs get detected and repaired during the early stages of development. Automated quality feedback assessments utilizing unit tests together with integration tests and static code analyzer SonarQube generate more stable programs that limit integration complications (Shahin, 2017). The deployment of Docker as a container solution enables developers to normalize development environments so that testing and production environments remain consistent, which reduces system compatibility issues (Bernstein, 2014).

Reliable software deployments occur through well-managed release processes, while software implementations can be performed without delay. Continuous distribution (CD) deploys software automatically without human intervention to eliminate delivery delays as well as reduce errors by using minimal operator interaction. Programmed scheduling and planning techniques minimize the dangers of problematic deployments by giving developers enough time for thorough testing and debugging processes. System reliability increases along with development issue identification speed through the application of defined rollback methods together with monitoring systems like Prometheus and Datadog.

These suggested changes will boost developer collaboration without generating deployment problems and secure the software infrastructure. The methodology draws its foundation from industry research to deliver a single procedure leading to effective results and dependable software development.

#### <sup>4</sup> **REFERENCES**

Cockburn, A., & Highsmith, J. (2001). Agile software development, the people factor. *Computer*, 34(11), 131-133.

<sup>3</sup>  
Ambler, S. W., & Lines, M. (2012). *Disciplined agile delivery: A practitioner's guide to agile software delivery in the enterprise*. IBM press.

<sup>5</sup>  
Bernstein, D. (2014). Containers and cloud: From LXC to docker to Kubernetes. *IEEE cloud computing*, 1(3), 81-84.<sup>1</sup>

Shahin, M., Babar, M. A., & Zhu, L. (2017). Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices. *IEEE access*, 5, 3909-3943.

<sup>2</sup>  
Pillai, B. (2023, November). Understanding Real-time Payment Dynamics in Australia. In *International Symposium on Distributed Ledger Technology* (pp. 124-145). Singapore: Springer Nature Singapore.

# SonamR\_A1\_ProjectProposal

## ORIGINALITY REPORT

10%

SIMILARITY INDEX

8%

INTERNET SOURCES

6%

PUBLICATIONS

9%

STUDENT PAPERS

## PRIMARY SOURCES

1

jyx.jyu.fi  
Internet Source

2%

2

Submitted to Flinders University  
Student Paper

2%

3

Ferreira, Mariana Moutinho Amorim.  
"Improving Organizational Documentation  
Practices: A Case Study of EFACEC.",  
Universidade do Porto (Portugal)  
Publication

2%

4

danielnmai.wordpress.com  
Internet Source

1%

5

vdocument.in  
Internet Source

1%

6

baft.org  
Internet Source

1%

7

Submitted to Southern Cross University  
Student Paper

<1%

Exclude quotes Off

Exclude bibliography Off

Exclude assignment  
template On

Exclude matches Off