PAyTEL

Marsol Daman, Lauren Lingeman, Axel Van Hoyweghen,
Alexander Dapoz, Anthony Calandra , Noah McGivern

CSI-4999-Senior-Design

Fall 2018

Outline

Project vision **5**
- Backgrounds 5
- Socio-economic Impact, Business Objectives, and Gap Analysis 5
- Security and ethical concerns 5
- Glossary of Key Terms 6

Project Execution and Planning **6**
- Team Information 6
  - Alexander Dapoz 6
  - Anthony Calandra 6
  - Axel Van Hoyweghen 6
  - Lauren Lingeman 6
  - Marsol Damon 6
  - Noah McGivern 6
- Tools and Technology 6
  - Android Studio, Java, Kotlin, and groovy 6
  - HTML, JavaScript, CSS, Bootstrap, and JQuery 6
  - Amazon Web Services (AWS) 7
    - Mobile hub 7
    - DynamoDB 7
    - Cognito 7
    - S3 7
- Project Plan 7
  - Sprint 1 (9/6 - 9/13) 7
  - Sprint 2 (9/13 - 9/27) 7
  - Sprint 3 (9/27 - 10/13) 7
  - Sprint 4 (10/13 - 10/25) 8
  - Sprint 5 (10/25 - 11/8) 8
  - Sprint 6 (11/8 - 11/22) 8
- Best standards and Practices 9

System Requirement analysis **9**
- Functional requirements 9
- Non-functional requirements 9
- On-screen Appearance of landing and other pages requirements 9
- Wireframe designs 9

Functional Requirements Specification **9**
Stakeholders 9
Actors and Goals 9
User Stories, scenarios and Use Cases 9
System Sequence / Activity Diagrams 9

User Interface Specifications **9**
Preliminary Design 9
User Effort Estimation 9

Static Design **9**
Class Model 9
System Operation Contracts 9
Mathematical Model 9

Dynamic Design **9**
Sequence Diagrams 9
Interface Specifications 9
State Diagrams 9

System Architecture and System Design **9**
Subsystems / Component / Design Pattern Identification 9
Mapping Subsystems to Hardware (Deployment Diagram) 9
Persistent Data Storage 10
Network Protocol 10
Global Control Flow 10
Hardware Requirement 10

Algorithms and Data Structures **10**
Algorithms 10
Data Structures 10

User Interface Design and Implementation **10**
User Interface Design 10
User Interface Implementation 10

Testing **10**
Unit Test Architecture and Strategy/Framework 10
Unit Test definition, test data selection 10
System Test Specification 10
Test Reports per Spring 10

Project Management **10**

Project Plan 10
SAME AS 2.3 (COPY OVER AT THE END) 10
Risk Management 10
References **10**

1.Project vision

1.1. Backgrounds

PAyTEL was originally thought of by the predefined project description given to us by Professor Patel. Here is the background given in the project description: “Credit card fraud isn’t new to any of us. Fraudulent use of credit cards is at all time high with growth of cyber retail of goods and services where identity check of the buyer is an issue. To cope with this, credit card issuers are constantly in search of better methods of identity check for online retail transaction. One recent addition to this system is SMS message based code verification as extra step for verifying buyer’s identity. In this project you will design a new system for Secure Transaction. The system consists of three pieces, a web portal that manages all transactions, a buyer mobile app for approval of transaction and a seller mobile app for issuing a charge. The front facing mobile phone camera will be used for authenticating biometric identity of the buyer when approving the transaction. Server will mediate the messaging, manage user profile and transactions.”

PAyTEL is a multichannel payment system for secure transactions for buyers and sellers.

1.2. Socio-economic Impact, Business Objectives, and Gap Analysis

The Socio-economic Impact for PAyTEL is going to impact the buyers and sellers in the economic marketplace and allow merchants and consumers to checkout and pay per transaction. It will impact the mobile app payment marketplace and allow more secure banking for mobile applications for merchants and consumers. Our business objectives are to allow merchants and consumers to have secure mobile banking as well and have the ability to issue charges as well as customize user profiles. Another business objective will be for merchants and consumers to search transaction history. Our desire for PAyTEL is to lead the mobile banking app industry to a new frontier, with clean, efficient, user interface, along with in depth user profile customization. The actual performance of PAyTEL will be determined by the volume of transaction requests created by merchants and consumers that find the application and utilize its full potential.

1.3. Security and ethical concerns

Our focus for PAyTEL will be fast but secure transactions for merchants and consumers. It will have encrypted data along the entire transaction process.

Ethically, we will not allow the average user to access any transaction apart from their own as well as limiting displayed personal information.

1.4. Glossary of Key Terms

2.Project Execution and Planning

2.1. Team Information

2.1.1. Alexander Dapoz

Front-end development and documentation.

2.1.2. Anthony Calandra

Application development and documentation.

2.1.3. Axel Van Hoyweghen

Lead back-end (server, database, etc.) design and implementation and documentation.

2.1.4. Lauren Lingeman

Application development and managing documentation.

2.1.5. Marsol Damon

Application development and documentation.

2.1.6. Noah McGivern

Front-end development and documentation.

2.2. Tools and Technology

2.2.1. Android Studio, Java, Kotlin, and groovy

Android Studio, Java, Kotlin, and groovy will be used to develop an Android application that will be used by the seller's and buyer's.

2.2.2. HTML, JavaScript, CSS, Bootstrap, and JQuery

HTML, JavaScript, CSS, Bootstrap, and JQuery will be used to develop the front-end of the transaction portal.

2.2.3. Amazon Web Services (AWS)

2.2.4. Mobile hub

Tool used to manage AWS services for this application

2.2.5. DynamoDB

NoSQL database

2.2.6. Cognito

Secure user authentication

2.2.7. S3

File storage

2.3. Project Plan

2.3.1. Sprint 1 (9/6 - 9/13)

- Define requirements
- Layout basic sprint schedule
- Setup collaboration tools (GitHub, Slack, Google Drive)
- Select development environment, languages, tools, etc. and get all team members access (Android Studio, AWS DynamoDB, Bootstrap, Java, Android SDK 8, Bootstrap, Groovy, Kotlin, Java JDK, CSS, HTML, Javascript)
- Begin project documentation

2.3.2. Sprint 2 (9/13 - 9/27)

- Setup AWS (Cognito, IAM, Mobile Hub) for user account creation and authentication
- Design consistent UI/UX for application
- Setup QA for testing for user authentication system
- Research and test phone facial recognition system for user authentication
- Design database for storing transactions and user information
- Write user stories for Sprint 3

2.3.3. Sprint 3 (9/27 - 10/13)

- Implement UI/UX for application (login, registration, user account page, main landing page)
- Connect application to login server and database storing user information
- Design UI/UX for transaction portal
- Setup transaction portal login at backend for admin

- Setup QA for testing transaction portal login
- Write user stories for Sprint 4

2.3.4. Sprint 4 (10/13 - 10/25)

- Implement UI/UX for application (settings, anything else unfinished)
- Implement UI/UX for transaction portal (login, main landing page, database queries)
- Finalize database design
- Setup QA for testing database queries and storage
- Write user stories for Sprint 5

2.3.5. Sprint 5 (10/25 - 11/8)

- Implement UI/UX for transaction portal (anything unfinished)
- Debug and testing for whole application
- Feature lock by end of this sprint

2.3.6. Sprint 6 (11/8 - 11/22)

- Continue debug and testing
- Finish final documentation
- Prepare final presentation and poster board

2.4. Best standards and Practices

3.System Requirement analysis

3.1. Functional requirements

3.1.1. User must be able to create a user account based on their Google login
3.1.2. User must be able to offer payment to another user (buyer role)
3.1.3. User must be able to accept payment from another user (seller role)
3.1.4. User must pass face scan to access their personal account
3.1.5. User must be able to see their transaction history
3.1.6. User must be able to change their payment method through the settings page
3.1.7. User must be able to generate a QR code
3.1.8. User must be able to scan a QR code
3.1.9. Admin must be able to access all transactions
3.1.10. Admin must be able to report malicious transactions to users

3.2. Non-functional requirements

3.2.1. User must have a valid Google login consisting of username and password
3.2.2. User must have a valid credit card
3.2.3. User must have an phone running Android API 26 or higher
3.2.4. Database must store user account information securely
3.2.5. Database must store all transactions
3.2.6. Server must be able to handle all incoming transactions real-time

3.2.7. Website and mobile app must both use the same color palette throughout.
3.2.8. Data must be encrypted when transferred between web-app / mobile-app and the server
3.2.9. Web-app interface must be responsive for variable screen sizes
3.2.10. Server must remove pending transactions that have timed out

3.3. On-screen Appearance of landing and other pages requirements
3.4. Wireframe designs
4.Functional Requirements Specification
4.1. Stakeholders
4.2. Actors and Goals
4.3. User Stories, scenarios and Use Cases
4.4. System Sequence / Activity Diagrams
5.User Interface Specifications
5.1. Preliminary Design
5.2. User Effort Estimation
6.Static Design
6.1. Class Model
6.2. System Operation Contracts
6.3. Mathematical Model
7.Dynamic Design
7.1. Sequence Diagrams
7.2. Interface Specifications
7.3. State Diagrams
8.System Architecture and System Design
8.1. Subsystems / Component / Design Pattern Identification
8.2. Mapping Subsystems to Hardware (Deployment Diagram)
8.3. Persistent Data Storage
8.4. Network Protocol
8.5. Global Control Flow
8.6. Hardware Requirement

9. Algorithms and Data Structures
    9.1. Algorithms
    9.2. Data Structures
10. User Interface Design and Implementation
    10.1. User Interface Design
    10.2. User Interface Implementation
11. Testing
    11.1. Unit Test Architecture and Strategy/Framework
    11.2. Unit Test definition, test data selection
    11.3. System Test Specification
    11.4. Test Reports per Spring
12. Project Management
    12.1. Project Plan
        12.1.1. SAME AS 2.3 (COPY OVER AT THE END)
    12.2. Risk Management
13. References