

QuikPAY Service Advisory Board Meeting August 18, 2022

1. Welcome introductions
 - Jody Brammel is new to the group
 - 13 attendees: Mark McCulloch, Joseph Muennich, Stephen Mosely, Angie Peatow, Cindy Nelson, Hannah Schneider-Lynch, Irina Botvinnik, Jody Bramel, Shaymond Michelson, Stacy Williams-Wright, Tamarra White, Tim Louis
2. Ecommerce basic elements and site tour
3. Ecommerce advantages
 - Fast turn-around
 - Automatic daily Banner deposit
 - No contracting or third party license fees
 - No merchant account needed
 - No in-house administration or development needed for business continuity
4. Quikpay API basic elements
 - Used when a custom payment interface is deemed beneficial or back end system integration is required, and the department has an available development resource
 - Now hosted on MS Azure instead of UO VM
 - LCB still transitioning
 - Transaction reporting by email file, or feed or on ecommerce.uoregone.edu
 - Will integrate with third party software such as a CRM
5. Third party integrations
 - Possible using the Azure API
 - For direct QuikPay integration Business Affairs can create custom QukPay “orders”
Examples:
 - Slate has direct QuikPay integration,
 - Register Blast (UO testing center)
 - InoSoft Fusion (PE and Rec)
 - StarRes (UO Housing)
6. Nelnet Service Charge Program
 - Can be used for education related payment processing
 - 2.3% service fee is collected from customer to cover card processing cost
7. Transaction reporting

Ecommerce has three reporting options

- Order reports show orders submitted with a specific payment form
- Line Item reports show the sales of given products/event registrations (independent of the originating payment form)
- Transaction reports show the raw stream of incoming transaction data delivered by QuikPay

Our API has separate reporting options but API transactions are also available on ecommerce reporting, they just don't have order information.

8. Refunds
 - Request submitted to UO Cashiers should include
 - Confirmation number
 - Amount to be refunded
 - Account holder name
 - Transaction date

9. Deposits

- Mike runs an Appworx job each morning (GWRQPAY) that processes the QuikPay End of Day file and deposits sales using index and account provided on Payment Card Acceptance request form.
- Same process updates any student accounts and Duck Bucks accounts with payments that were not processed in real time.

10. Card processing fees

- Negotiated with US Bank/Elavon
- Roughly 2% of transaction value
- Actually comprised of three components:
 - Card association fees that go to card brands
 - Interchange which goes to issuing bank (this is the largest component and vary depending merchant category and type of card presented), and
 - Flat 5 cents which goes to Elavon our processor.

11. Flywire international payments

- Flywire enables international wire payments at a low cost
- Also offers credit card payment with service fee
- Contact Krista Borg if interested

12. Nelnet Business Solutions update

- Mark is on NBS client advisory board
- Have new head of product innovation
- New third party payment product
- Campus Key virtual ID card product is a tough sell on campus because there are so many services that ride on plastic. The Campus Key app now in use at a few schools features:
 - virtual photo ID,
 - student account statement and payments,
 - declining balance account (DuckBucks),
 - blue tooth building access
 - meal programs
 - rewards programs
 - student and group messaging

13. Payment Card Acceptance Request is how departments get started. It is on the forms site under ecommerce.

14. Drupal 7

- Extended support through 2025
- Discussed D9 development opportunities and challenges

15. Enhancement requests, none noted.

16. Pain points

- Reporting has been more reliable this past year.
- We had a card verification fraud incident where Olum received a few thousand one cent payments. NBS experienced this with several clients. NBS remediated using firewalls, throttle and CAPTCHA.

17. Open Discussion

- EMU or Cassowary ?
- QR code use
- Northern lights