

# Improving The Web Services for Remittance Company: Express Remit as a Case Study

Hewa Majeed Zangana<sup>1</sup>

<sup>1</sup>Department of Information Technology, Technical College of Administration, Duhok Polytechnic University, Kurdistan Region of Iraq

## ABSTRACT

Money transfer, often known as remittance, is the process of a foreign worker sending money back to his or her home country. For centuries, remittance services have played a critical role in the economics of numerous countries. Express Remit is a Malaysian remittance company that provides remittance services via wire transfer and electronic network system, with the goal of becoming one of the world's leading remittance service providers. The company's processing system is highly fast and secure, but the data integration between the company's branches and its correspondent banks and agents has to be improved and fully connected. This paper will aid us in determining whether the web services we intend to install in the organization will aid in data process integration.

**Keywords:** *Express Remit, Remittance, Wire Transfer, Electronic Network System.*

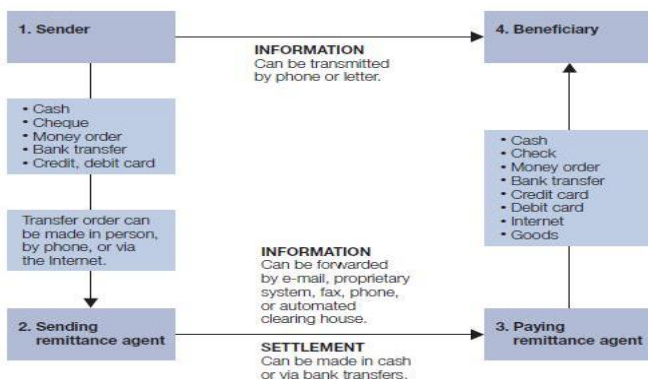
## 1. Introduction

Remittance flows have become a more global and vital source of development money for poor countries in recent years. Remittances from abroad have surpassed development assistance and are now second only to foreign direct investment and capital flows to underdeveloped countries. Remittances are the main source of income for developing countries' economies as a whole, as well as for millions of individual households, notably for poor women and their children. They directly benefit the poor, unlike aid or private investment flows. It also provides remittance services as a way for financial institutions to promote awareness about the importance of remittance to underprivileged clientele. (.NET) Checks, bank transfers, and ad hoc services like Western Union, MoneyGram, and Vigo are all examples of formal financial system mechanisms used to send remittances. (An illustration of the transfers process is shown in Figure 1). (Agrawal)

Remittance is a multi-tiered web-based client/server remittance software with several add-on modules. Users can read their reports in a variety of currencies, including local currency, using the currency converter feature. The charge that customers must pay for each transaction is known as remittance service. This research will help us decide whether the web services we plan to implement in the organization will help with data process integration.

## 1. BACKGROUND OF STUDY

In this study, the company we chose used a complete remittance solution that is used by banks, development banks, and financing and remittance organizations. The system is user-friendly, simple to maintain, adaptable to user upgrades, and, most importantly, completely secure. The company employs industry-standard technologies, including an ASP/PHP HTML/java script front-end (GUI interface) and an MS SQL 2000 server back-end (database). (Remittances: strategic and operational considerations, 2006) Later in the proposal, the system's hardware, software, and other secure tool requirements will be detailed. The SDLC paradigm will be followed by the Express Remit money transfer software. The development team will go through all of the needed phases during the development process, which are as follows: -



(Figure 1: General Remittance)

- **System/Information Engineering and Modeling**

A fast and enhanced money transfer system is the result of a lengthy process that begins with the definition of requirements for all system elements and then assigning a subset of those criteria to software. When software must interact with other aspects such as hardware, people, and other resources, a system perspective is required. The existence of software in any entity is contingent on the existence of a system. In some circumstances, the system should be re-engineered for maximum output. (Narang, 2020)

- **Software Requirement Analysis**

The development team will visit the customer and study their system requirements during this requirement analysis phase. In the supplied software system, the development team will look at the need for feasible software automation. (Meyer, 2019) The development team will then deliver a paper containing all of the candidate system's particular recommendations.

Software is the focus of the requirements analysis and information gathering process. (Orozco, 2013) The development team will also collect any other information that may be required for the software update. The team will investigate the software's information domain as well as the needed function, behavior, performance, and interfacing at the end of this phase.

- **System Analysis and Design**

The entire software development process, as well as the overall program structure and outlay, will be determined during this phase. This phase will determine the number of tiers required for the package architecture, database design, data structure design, and other aspects of client/server processing technology. (Migration and remittances: Recent developments and outlook – Brief, 2019) It plays a crucial role throughout the entire development cycle. During this phase, the development team will exercise extreme caution:

- a) The whole design will be created according to their Analysis.
- b) The logical system of the software will be developed.

- **Code Generation**

In Code Generation phase, the design will be decoded into a machine-readable form. For generation of code, Programming tools like Compilers, Interpreters,

and Debuggers will be used. For coding purpose PHP/ASP, HTML/XHTML, XML, will be used. (MySQL Database solution) As for the platform, the XAMPP/WAMP will be used. (Tiroler Volkskunstmuseum, 2017) The development team will choose the appropriate language which one is suitable for the process.

- **Testing**

After code generation phase the software testing will begin. Different testing methods will be applied to detect the bugs that could be committed during the previous phases. (Waldinger, 2015) A number of testing tools and methods are required for testing purpose.

- **Maintenance**

When the Express Remit Money Transfer System is handed to the organization, it, like any other system, will undoubtedly undergo changes. (Page, 2012) There is a plethora of explanations behind the shift. Change could occur as a result of unexpected input values into the system. Furthermore, changes in the system have a direct impact on the software operations. (Silver, 2015) The mechanism will be put in place to accommodate any changes that may occur during the post-development phase.

## 2. PROBLEM STATEMENT

ExpressRemit is a remittance provider that sends money to practically every country in South Asia and Southeast Asia. As previously said, remittance companies send money through banks, hence banks are an important element of the remitting process. Banks do not always provide banking services to all remittance companies due to regulation and agreement concerns. The remittance companies then rely on one another. If a consumer wants to send money to a bank that does not work with ExpressRemit, they must use one of the other remittance firms that works with that bank. The issue is that it's tough to keep track of transactions in those situations.

Currently, other remittance businesses send an MS Excel or CSV file, or sometimes just a printout, of all the transactions made by ExpressRemit at the end of the month. After that, ExpressRemit's employees manually enter the transactions into the database. This takes up a lot of time, money, and human resources. It's also difficult to gain a sense of the company's overall status in the middle of the month.

## 3. STATEMENT OF PURPOSE

The goal of the project is to create a web application that will replace the present application and add advanced features to the system to address the issues

mentioned previously, such as system integration and instant messaging. ExpressRemit solution is the name of the application. The ExpressRemit solution promises to deliver a user-friendly and standardized enterprise web app. The ultimate goal is to create an API for ExpressRemit so that other remittance firms can simply interface their systems with ExpressRemit and ExpressRemit can integrate with other remittance companies as well as financial institutions such as banks.

**3.1 User Friendly Interface** - ExpressRemit solution is promised to build an easy-to-use web-interface for Administrators, Tellers, Payout Agents and Other Finance Institutes (e.g., Banks and other Remittance company).

**3.2 Easy Import and Export of Data** - Daily, Weekly, Monthly and annual report will be easy to import to and export from one system to another.

**3.3 API for system integration** - an application programming interface for other remittance and finance institute to have their system integrated with express remit.

**3.4 Instant Messaging** - IM for instant communication between branches and other institutes. This is an extra feature just to speed up the process of communication and to send notification to a particular group or to broadcast message across the branches of ExpressRemit.

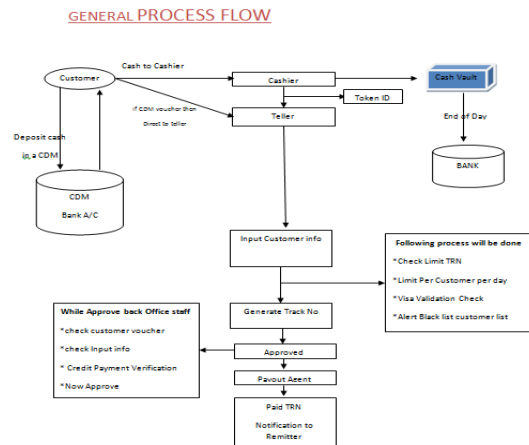
**4. CURRENT ARCHITECTURE**

ExpressRemit currently uses third-party remittance software developed and managed by a Software Solution Provider. As a result, some functionality, such as administrator account management, are only available to third-party solution providers. The database is hosted by a third-party hosting service. A digital certificate encrypts all communication sessions. The following are some of the app's users:

- Administrator
- Teller
- Payout Agent
- Partner

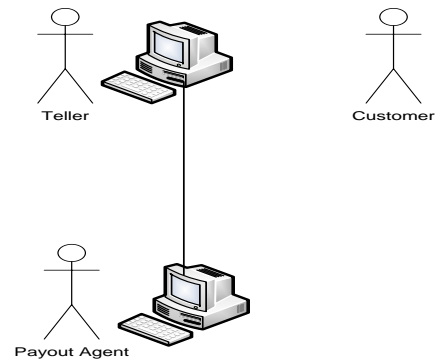
Administrators are users who are in charge of overseeing the local network as well as any other IT-related activities. They are in charge of the solution, as well as the creation, management, and control of various sorts of users, roles, and accesses. A teller takes the customer's money and enters it into the system for future reference. Other financial institutions or banks could act

as Payout Agents. Other remittance companies and financial institutions are partners. There are two sorts of partnerships: one in which ExpressRemit sends money to other remittance firms, and the other in which other companies send money to ExpressRemit. The monthly report of transactions conducted through others is collected in two ways. Some partners send/upload monthly reports to ExpressRemit's website, and others provide an interface through which other remittances can view and print transaction reports generated by that remittance.



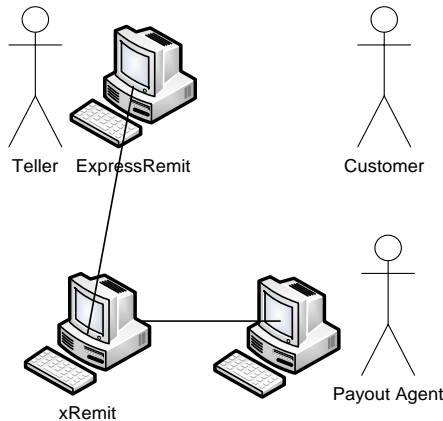
(Figure 2: General Process Flow)

The remittance procedure begins when a customer pays cash to a teller at the counter, who then receives the money, enters the customer's information and the bank name from where the money will be received into the system via a web-interface, and gives the customer a pin number. Once the transaction has been entered into the system, it will appear in the Payout Agent interface, which in this example is a bank. The bank sends the money to the beneficiary, who then gives the bank the pin number. The payment is marked as done/successful once it has been completed. At any moment, payout agents can access transaction history and export the report as an excel file.



(Figure 3: Agent, Teller, and Customer)

If a customer wishes to receive money from a bank or finance institute that does not work with the remittance firm, the teller must utilize a different interface supplied by another remittance company that works with that bank. Furthermore, the teller must enter transaction information into two separate interfaces: one from ExpressRemit's teller web interface and the other from the external remittance firm.



(Figure 4: Agent, Teller, Customer, and ExpressRemit)

For the time being, ExpressRemit does not have an integration mechanism in place to track transactions handled by external remittance. ExpressRemit must request the transaction report, which is typically delivered in the form of an Excel or CSV file, and administrators must manually enter the data into the system. This is inefficient and wastes a lot of time.

### 5. PROPOSED ARCHITECTURE

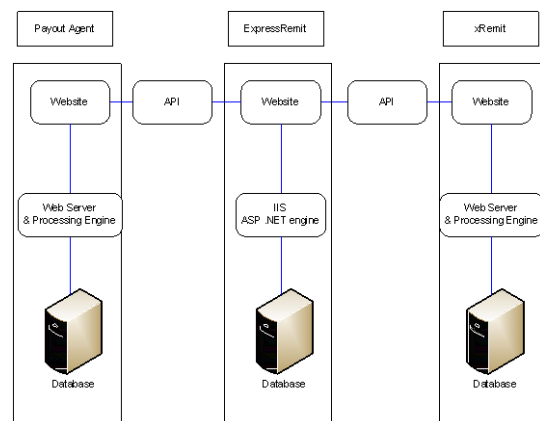
The recommended approach is to rewrite the current program with some interface changes and extra functionality, primarily to implement the integration mechanism. Instant Messaging and Message Broadcasting are two additional functions. The users would be identical to those of the current system; the only variation would be how they interacted with it. Currently, an external user must have a different role and login credentials in order to access Express Remit data. Other remittance companies will be able to integrate their applications with ExpressRemit's once the new system is in place. Some processes will be automated, and data will be easier to interchange between different systems or platforms.

### A. Components of Proposed system

- Web Server – Microsoft’s Internet Information Service will be used to run the web application from a windows environment.
- IM – a light weight messaging application which will be embedded in the web site to allow the user communicate in case of emergency.
- Database – MySQL database server will be used to store all system and customer related information, transactions and event logs.
- Web Interfaces – For the user to make use of the overall system four different web-interfaces will be built.
- API – a key which is to authenticate remote application to give access to the ExpressRemit solution to access certain predefined areas.
- Digital certificate – to ensure secure connection.

### B. Proposed solution will have four different Web-Interfaces:

- Administrator’s Interface
- Teller’s Interface
- Payout Agent’ and
- Partner’s Interface



(Figure 5: Administrator’s Interface)

Administrators can utilize the Administrator's Interface to create, administer, and control user accounts, roles, and privileges, as well as perform complex operations such as blocking transactions,

altering transactions as needed, deleting transactions, and blocking users.

Tellers are the front-line employees who interact with customers, register them, and guide them. They are in charge of collecting data and storing it in the system.

The money is normally paid out to the opposite end by payout agents, which are usually banks. Payout agents will be allowed to examine and update some fields of new incoming transactions, but will only be able to view the history of previous transactions. If any of the partners does not have the capacity to integrate with ExpressRemit, they can use the Partner interface to create new transactions and examine existing ones.

## 6. PROPOSED PLATFORM

The ExpressRemit Solution application will be developed on the ASP.NET framework, and the backend database system will be MySQL community edition. ExpressRemit will be integrated with other remittance solution applications using an XML web service.

6.1 **ASP .NET** - is a web application framework developed by Microsoft. ASP .NET has a rich set of libraries which allows developers to build rich dynamic web sites, applications and web services [3]. In this project ASP .NET is used to build web-interfaces.

6.2 **C# .NET** - an application development platform. C# is used to enhance functionality of ASP .NET components and to provide rich user experience and interaction with the web interface.

6.3 **MySQL** - an RDBMS (relational database management system) owned by Sun Microsystems. (Jan Wimaladharma, March 2004) MySQL has several additions. In our case we will be using MySQL enterprise edition.

6.4 **XML** - Extensible Markup Language (XML) is a general-purpose information storage system, a markup toolkit. XML is an open standard all most all the programming languages and platform. It's common standard.

## 7. MANAGEMENT AND TECHNICAL CONSTRAINTS AND RISKS

### 7.1 Management constraints:

- Software systems design: the proposed system is using ASP.NET and C#. Instead on that the implementation done by using PHP because of

two reasons. Firstly, for exiting web application used both of platforms (ASP.NET and PHP). Secondly, many errors raised in building process by ASP.NET and limited time to implement the proposed solution.

- Time: Because of limited time, building a web service by using ASP.NET may have to take longer time than php to get the same result. On other hand, achieving the goals from this propose and builds the web service which contain the developing area to meet the user requirement is the main issues.

### 7.2 Technical constraints:

- Functionality Overlap: The existing system implements same functionality across the agents and remittance partners. Therefore, there's a functionality overlap in the system.
- Data Duplication: Current system does not allow one remittance's web application to get data from another, instead the share collected data for a period using Excel or CSV file. In that case, ExpressRemit has to insert those data manually to their system, which leads to data duplication.
- Different Semantics: Each remittance house has their own web-application made by different vendors to handle their transaction and other remitting processes. In most cases, it confuses the user of the system, as each vendor has their own way defining a term. For example, one remittance house may say "Pay-In" agent and other could say "Sending" agent, both terms have same meaning; however, it could confuse the user.

## 8 FINDINGS

### 8.1 Business issues:

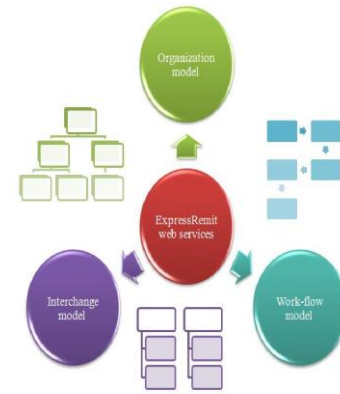
1. The proposed solution is targeted for data integration between remittance web-applications. These data could be transaction details, status of a transaction, daily or monthly report etc.
2. ExpressRemit's teller enters transaction details containing sender details, beneficiary's details, and payout agent's details. If the transaction needs to be processed by a partner of ExpressRemit, while adding the transaction the teller specifies the partner. Partners are basically other remittance companies who

have joined with ExpressRemit to extend their service. Partner can make http request to check if they have any transaction to process. The results returned as xml file, which the partner can process anyway the want, as xml is readable by all programming language from any platform. Once transaction details are retrieved from remote system (ExpressRemit), they can use any xml parser and represents the information in their native standard for payout agent.

Transaction Processing	
Payout Agent	Process Transaction
Partner	Submit Request to check whether any transaction is to be processed
Teller	Enter Transaction Details

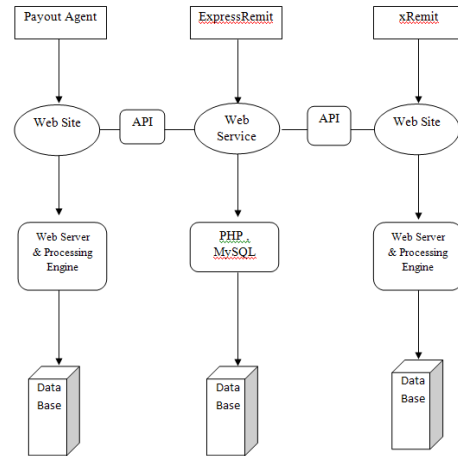
(Figure 6: Transaction Processing)

- Partner’s web application requests for pending transaction. If there’s any pending transaction an XML file is generated and the link is returned to the application. And xml parser can parse the data from that xml and manipulate parsed data as needed.
- Framework: the framework divided into three area as a following:



(Figure 7: Framework)

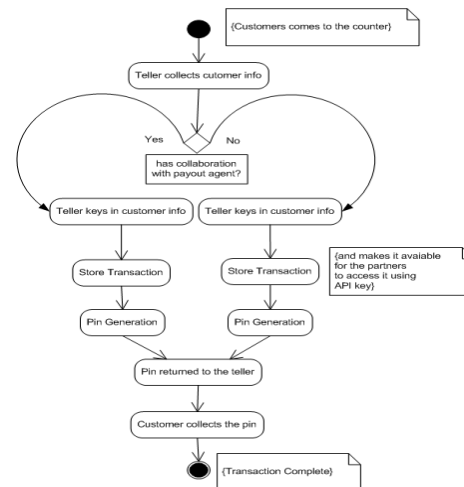
### 8.2 Organization model:



(Figure 8: Model)

### 8.3 Work-flow model:

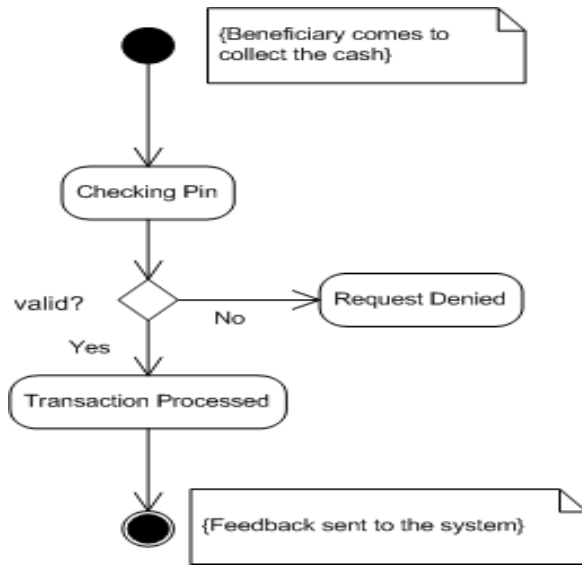
- Express Remit Work-flow



(Figure 9: Work-flow)

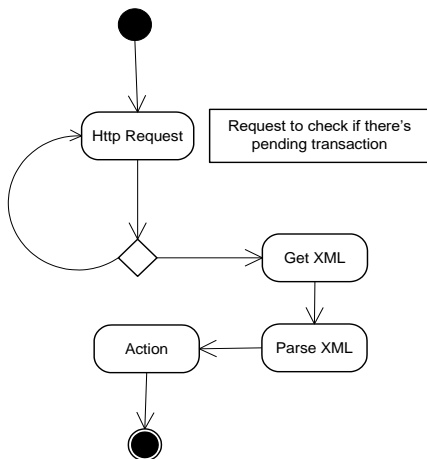


2. Payout work-flow



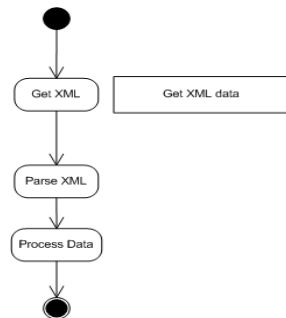
(Figure 10: Payout Work-flow)

3. Partner's web application wok-flow



(Figure 11: Application work-flow)

8.3 Interchange



(Figure 12: Interchange)

8.4 Technical issues:

1. Technical interoperability:

Current web-application does not allow external agents to get data from the system. There are no means to get and insert external data automatically into the system. In addition, they don't have a way to send or offer transactional information to third-party remittance house. The proposed solution utilizes XMLHTTPREQUEST to get third-party data and JQuery as XML parser to parse the data.

2. Utilizing XMLHTTPREQUEST now all any system can request for transaction related information from Express Remit and also Express Remit now has the ability to parse external data automatically.

3. JQuery is a JavaScript frame work which has built in AJAX functionality. AJAX makes it possible to make request in the background without disturbing the user, which means user do not have to make the request manually, the request will be posted automatically with the help of XMLHTTPREQUEST.

4. The integration happens in application layer, where data are exchanged between web-services. Data are interchanged in XML format with is known for universal interchangeable language. XML data can be parsed by any programming language, and it's a standard way of sharing data. Organization participating this change in Express Remit shall be willing to adopt with the change.

9 CONCLUSION

Remittance services have played an important part in the economies of many countries for millennia. Express Remit is a Malaysian remittance firm that offers wire transfer and electronic network system remittance services, with the goal of becoming one of the world's largest remittance companies.

Express Remit's solution which provides the build of the web application. It will add advanced functionality to the system to be more powerful and integrated. In addition, the main objectives of the Express Remit solution are provided User Friendly Interface, Easy Import and Export of Data, and IM for instant communication between branches and other institutes which primes to makes the business

process more efficiency and fully integrated.

## REFERENCES

.NET, A. (n.d.). *a web application framework*. Retrieved from Wikipedia: Wikipedia.org

Agrawal, A. (n.d.). *Business process mapping of enterprise integration Infosys:Winter2005*. Retrieved from [www.infosys.com](http://www.infosys.com).

Betts, J. G. (2020). Text expanding on the checklist for proper reporting of evidence in sport and exercise. *International Journal of Sport Nutrition and Exercise Metabolism*, 2-13.

Jan Wimaladharma, D. P. (March 2004). Remittances: The New Development Finance? *Small Enterprise Development Journal*.

MySQL Database solution. (n.d.). Retrieved from [www.sun.com/software/products/mysql](http://www.sun.com/software/products/mysql)

Narang, B. A. (2020). A tool to explore discrete-time data: The Time Series Response Analyser. *International Journal of Sport Nutrition and Exercise Metabolism*, 374-381.

*Remittances: strategic and operational considerations*. (2006). IFAD "International Fund for Agricultural Development".

Meyer, S. (2019). *Remittances und transnationale Kapital* [Remittances and transnational capital].

In K. Braun, C.-M. Dieterich, & J. Moser (Eds.),

*Wirtschaften. Kulturwissenschaftliche Perspektiven*

[Economic practices. Anthropological perspectives]

(pp. 64-80). Marburg: Makufee.

World Bank Group. (2019). *Migration and remittances*:

*Recent developments and outlook – Brief 31*. Washington,

DC: World Bank. Retrieved from [https://](https://www.knomad.org/publication/migration-anddevelopment-brief-31)

[www.knomad.org/publication/migration-anddevelopment-](https://www.knomad.org/publication/migration-anddevelopment-brief-31)

[brief-31](https://www.knomad.org/publication/migration-anddevelopment-brief-31)

Tiroler Volkskunstmuseum. (Ed.). (2017). *Hier Zuhause*.

*Migrationsgeschichten aus Tirol* [At home here.

*Migration stories from Tyrol*]. Innsbruck: Tiroler

Landesmuseum.

Waldinger, R. (2015). *The cross-border connection: Immigrants,*

*emigrants, and their homelands*. Cambridge,

MA: Harvard University Press.

Silver, H. (2015). *The contexts of social inclusion* (UNDESA

Working Paper No. 144). New York, NY:

DESA. Retrieved from [https://www.un.org/esa/desa/](https://www.un.org/esa/desa/papers/2015/wp144_2015.pdf)

[papers/2015/wp144\\_2015.pdf](https://www.un.org/esa/desa/papers/2015/wp144_2015.pdf)

Orozco, M. (2013). *Migrant remittances and development*

*in the global economy*. Boulder, CO: Lynne Rienner

Publishing.

Page, B., & Mercer, C. (2012). *Why do people do stuff?*

*Reconceptualizing remittance behaviour in diasporadevelopment*

*research and policy*. *Progress in Development*

*Studies*, 12(1), 1-18.