Nour Khatib
S-DB-IPS3-S3-DB01

441295@student.fontys.nl

SN: 3962881

LO 1- Web application - GP + IP:

This part is done yet.

LO 3- Agile method - GP:

Agile

Agile development provides a guideline, a blueprint, and a way of thinking which can help greatly a software developer when making choices whether before-during or in the last stages of development.

Agile provides a set of values that could guide the way of thinking of the developer each time the developer needs to turn into a new task, or even when iterating over previous tasks.

To be more clear, agile values shows what is important and declare clear importance for:

- Individuals and interactions over processes and tools.
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation; and.

- Responding to change over following a plan.

These clear values came out of the need for a way of working that is not just more efficient but more humane as well, as software development changes is such a raped speed, a need of w working mentality that emphasis the importance of the human/PO/Developers is needed.

Agile is the new phase of consideration needed for modern development, where it guides it by clear blue-print of values and principles that ease choice making process for all the stakeholders of the development.

Agile provides 12 principle that comes hand in hand to elaborate and extends over the values

1. Satisfy Customers Through Early & Continuous Delivery.
2. Welcome Changing Requirements Even Late in the Project.
3. Deliver Value Frequently.
4. Break the Silos of Your Project.
5. Build Projects Around Motivated Individuals.
6. The Most Effective Way of Communication is Face-to-face.
7. Working Software is the Primary Measure of Progress.
8. Maintain a Sustainable Working Pace.
9. Continuous Excellence Enhances Agility.
10. Simplicity is Essential.
11. Self-organizing Teams Generate Most Value.
12. Regularly Reflect and Adjust Your Way of Work to Boost Effectiveness

To see how Agile can be of a direct help lets compare three of the methodologies guided by the agile values and principles, we will compare the main advantages of them and the disadvantages and give a recommendation what is best.

Firstly SCUM (Systematic Customer Resolution Unraveling Meeting)

SCRUM is:
“Scrum is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value”(www.scrum.org)

SCRUM is mostly used as a project development tool, where its routine and rituals shows how deeply the agile values and principles rooted in its guidance. SCRUM plans shorts sprint declaring user stories from the point of view of the user to provide a delivery that can directly add value after each sprint.

SCRUM impress synchronizing the team, and high level of communication and collaboration, as for each day a standup meeting is hold to give the direction for each member of the team what to work on, in addition that each member is knowing who is doing what, not to forget, that at each end of a day a small update meeting called stand down is hold to finalize the synchronizing of the work day.

SCRUM plans sprints with collaboration of the PO in the sprint review and adapt to the PO sprint review and peer review after each sprint at the sprint retrospective, which makes quit a nice case representing for the agile way of working. SCRUM emprise each sprint feedback, adapts to it, and reflects it through the product backlog and the delivered valuable product which is directly relevant the feedback review of the stakeholders and all in short terms.

SCRUM reflects its vision and direction of the product through the backlog, where the PO can always give feedback around each pull, at the end of each sprint, that feedback will be integrated at the sprint meeting and it will be visible in the backlog for the next pull .

Secondly Kanban

according to Wikipedia(https://en.wikipedia.org/wiki/Kanban_(development)):

”**Kanban** (Japanese: 看板, meaning signboard or billboard) is a lean method to manage and improve work across human systems. This approach aims to manage work by balancing demands with available capacity, and by improving the handling of system-level bottlenecks.”

As we compare the basic definition, we realize the similarities between SCRUM and Kanban are deeply rooted, which is of course the most expected as both are agile methodologies.

However agile provides guidelines not a rigid block of road that is forceful to work according, with that been said its is important to realize that the differences between SCRUM and Kanban is of a value.

“Kanban **helps visualize your work, limit work-in-progress (WIP) and quickly move work from "Doing" to "Done."** Kanban is great for teams that have lots of incoming requests that vary in priority and size. Whereas scrum processes require high control over what is in scope, kanban let's you go with the flow.” (atlassian.com)

These differences can be seen in how the rituals and routines of Kanban holds for example

SCRUM	Kanban
Scrum Teams work in a series of Sprints of 1, 2 (most common), 3 or 4 weeks duration.	Kanban is a continuous process. (cf. Scrum’s periodic Sprint.)
It is the job of the Scrum Master to help the Product Owner, the Development Team to develop and maintain good habits.	It is the job of the Agile Coach (if present - not all Kanban teams have one) to help the Product Owner and the Development Team to develop and maintain good habits.

Each Sprint starts with a Sprint Planning Meeting - facilitated by the Scrum Master and attended by the Product Owner and the Development Team and (optionally) other Stakeholders. Together they select high priority items from the Product Backlog that the Development Team can commit to delivering in a single Sprint. The selected items are known as the Sprint Backlog.	Items are “pulled” directly from the Product Backlog.
The Development Team works on items in the Sprint Backlog only for the duration of the Sprint. In all but exceptional circumstances, new issues must wait for the next Sprint.	Each column has a strict Work in Progress (WIP) limit. The WIP limits ensure that items move across the board in the shortest possible time.
The Daily Scrum (aka Daily Huddle, Daily Standup) is a short standup meeting attended by the Scrum Master, the Product Owner and the Development Team.	An empty - or nearly empty - column is a signal to the previous column to send another item. This is the “pull” system in action
A review of the Sprint. Often includes a demo of new features to Stakeholders.	The Daily Standup is a short standup meeting attended by the Agile Coach, the Product Owner and the Development Team.
An examination of what went well, what could be improved, etc. Aim: to make each Sprint more efficient and effective than the last.	Each item is packaged for release as soon as it is ready.
At the end of the Sprint, completed items are packaged for release to live. (Note that some teams release more often than this.) Any incomplete items are returned to the Product Backlog.	A demonstration of new functionality to Stakeholders
	A look at what went well, what could be improved, etc. Aim: to improve the process.

(“DevelopmentThatPays.com”)

Thirdly and lastly Extreme Programming:

“Extreme programming is a software development methodology intended to improve software quality and responsiveness to changing customer requirements” (Wikipedia)

“is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development” (agilealliance)

The XP is somewhat considered one is the closest methods to the agile aim, as it is manifested by overlap between agile and XP values.

XP five values:

- Communication.
- Simplicity.
- Feedback.
- Courage.
- Respect.

Advantages	Disadvantages
Close contact with the customer	Additional work
No unnecessary programming work	Customer must participate in the process
Stable software through continuous testing	Relatively large time investment

Advantages	Disadvantages
Error avoidance through pair programming	Relatively high costs
No overtime, teams work at their own pace	Requires version management
Changes can be made at short notice	Requires self-discipline to practice

(ionos)

Conclusion

LO 5- Cultural differences and ethics - GP:

Introduction

According to oxford dictionary the term itself contains layers and layers of description. Culture is wide multi aspect construct, that is layered and widened up and narrowed down like the layers of an onion.

When browsing the results coming when searching the original definition of what is culture, the first observation is how many distinguished types of cultures there are. Work culture, communication culture, religious culture of a certain group, national culture, family culture and so on and so on.

The different aspects of culture only show one dimension of the construct, as the reality is that for each distinguished type of culture many subcultures branching out of it.
For example when talking about working culture, we can compare flat or hierarchical way of management, how ever that does not include the devotion to work culture aspect neither the equality or the money transparency aspects.

In the previous example I only diverted from one major aspect of culture to some of the subcultures that could come out of it, however a really important adding is that it does not include the level of it, e.g. there are country working culture, company, mangers, workers, and all the micro unique special cases cultures.

Culture differences

I have come from and eastern culture (Syria) and have lived Dubai, Turkey, and lastly the Netehrlands. I have always loved to work with other cultures and learn how to

emphasis with others. I have studied one-year European studies at HBO Zuid in maastricht which taught me a lot about culture differences.

Culture differences can vary from communication to moral and ethical aspects.
each group collective behavior can turn into culture difference, e.g. the most bazar one is how intense the difference in communication between the political studies and the ICT study. The political studies way of communication is based on win or lose, with high emphasis on who own the negotiation (“even that sometimes it was just a discussion it turned to be a bargaining somehow”), however the communication with ICT students is more collaborative and focus more on filling the gaps and utilizing the individual talents in the group to optimize for a win/win situation.

Culture differences can vary from two countries cultures to two communities or schools, tow neighborhoods, and sometimes two neighbors where each has a different family culture.

My culture:

I have thrived and I hope that I will continue to thrive of collecting many small fragments of different cultures within me.
I consider my self an inbetweener that can bridge some aspects of the Levant in the east with some of the west of Europe, adding to that, I have spent ten years changing countries since the war of syria started, which mostly I have spent within expats communities, which by itself is a global/diverse/rich/colorful culture that I belong partially to as well.

Communications with respect to cultural differences in the group

I found it really important to set from the beginning a list of expectation, so I have asked to have a meeting where each one of us declared his expectation of how much effort/communication/semester-grade is expecting. After that I have made sure that I communicate personally with each one individually, which gave the time and the opportunity to create a common understanding of the person, chosen terms and vocabulary and the level of directness. After that I have made sure that when I can I will try to highlight the understanding that anyone can object and that we will only move forward within reasonable unanimous agreement.

The ethics of the software development

introduction: with more and more awareness about the enormous impact that software has on our lives, the topic to consider the moralities of our applications and the applications tactics is rising up and becoming a more needed consideration by the day.

In the last 20 years companies like Cambridge Analytica and Facebook, have been able to shape our world, many(including myself) will say not to the better but to the worse.

When software companies start to use the freedom of speech and exploit it with freedom or reach, we are never more exposed for foes more acknowledged and aware of our weaknesses than ever before, and reaching us inside our safe low guarded homes.

Not only private data companies are the threats, many shouts are calling for how easy it is for a handful of software engineers to switch governmental traffic sensors into monitoring surveillance agents.

The cases to prove the need to consider the morality and ethics of software development are endless and for the near future might(unfortunately) only to increase.

Ethical aspects in my work

The ethical considerations related to the projects I am helping developing needs to stretch for what beyond the right now level of development, as it is too easy to overstep some consideration if it was not made out of clear rules, (black and white) type of rules.

For the GP development we are working with PO that has its own agenda, like profit as a driving force for example. That could lead for the PO to ask for cross selling or up selling technics, through software solutions, an unethical approach will lead for increase in the consumerism behavior of the user without any consideration for sustainability for example and the balance between booth is not an easy task.

For my IP project I am developing a marketing/management system that aims to help small-medium business to compete with the giants' monopolies of the market.
However, to be able to compete with the monopolies of the market I might need to use some of their tactics.
Here come the question, when is it okay to use a gun(when the user consider it righteous aim to use it).
For example, when is it okay to use marketing strategies that increase consumerism aiming to switch costumers from Amazon to local business?

I find the need to develop an ethical guideline and consideration is essential like never before at the moment, I enjoyed working and reading from TICT, sapiens and daytimes, the many articles related to the potential power of AI.

LO 6- Requirements and Design – GP:

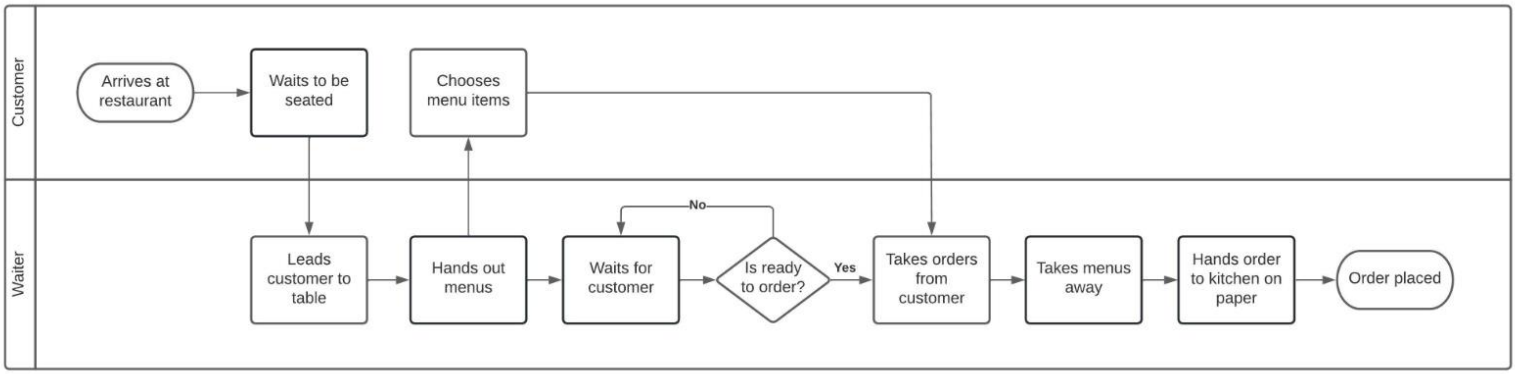
For the requirement and design, I started with Borga with library research and SOWT analysis that we included in the project plan, after that I went into the FoodHalen Amsterdam where they were using a similar system to the one asked by the PO and did field research.

I have asked some of the selling points in the Halen and the cashier and waiters about their experience with their system, after that, I have shared my findings with the team and together created a proposal design for the PO.

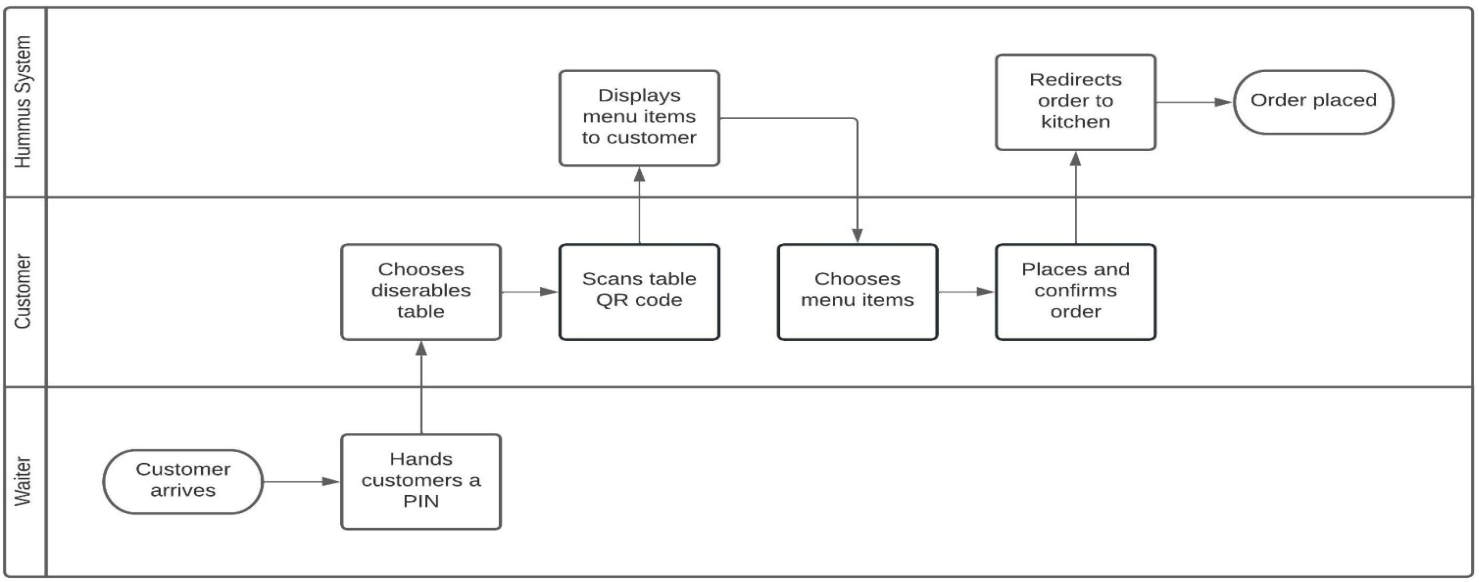
As we are working with scrum, the sprint review with PO always corrected and updated the acceptance criteria, that we iterated over our design sometimes and communicated with the PO through the email by the contact person Chandler during the sprints to check if our criteria is desirable by the PO.

LO 7- Business processes – GP:

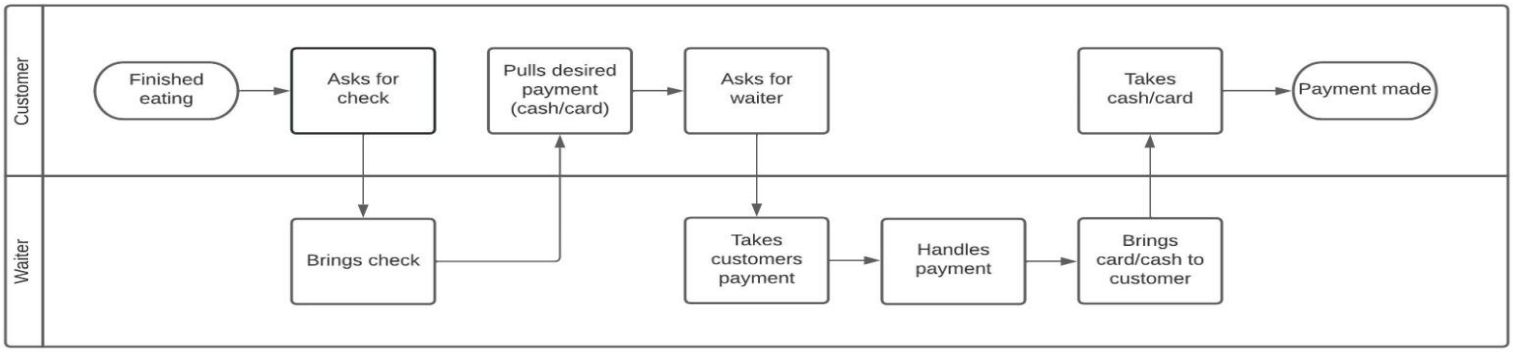
1.We have received a development request from Mediaan Heerlen as the product owner (PO). The product requested is about modernizing the ordering system within the restaurants serving industry.
Below is the typical manual way of ordering, and the PO is looking to automate this process to increase efficiency and win over the bottle necks, like the dependency on waiting between the waiter and the customer.



2.After sprint zero, where several meetings with the team, the PO and the coach a new system was proposed, with working flow for it diagramed below. The new system still functions for ordering in restaurants, but it automates and take over as much tasks as the PO allowed. you can realize for example that the there is no more waiting time for the waiter on the costumers to order and vis versa, the orders directly communicated to the kitchen, and the waiter is dispatched for different tasks.

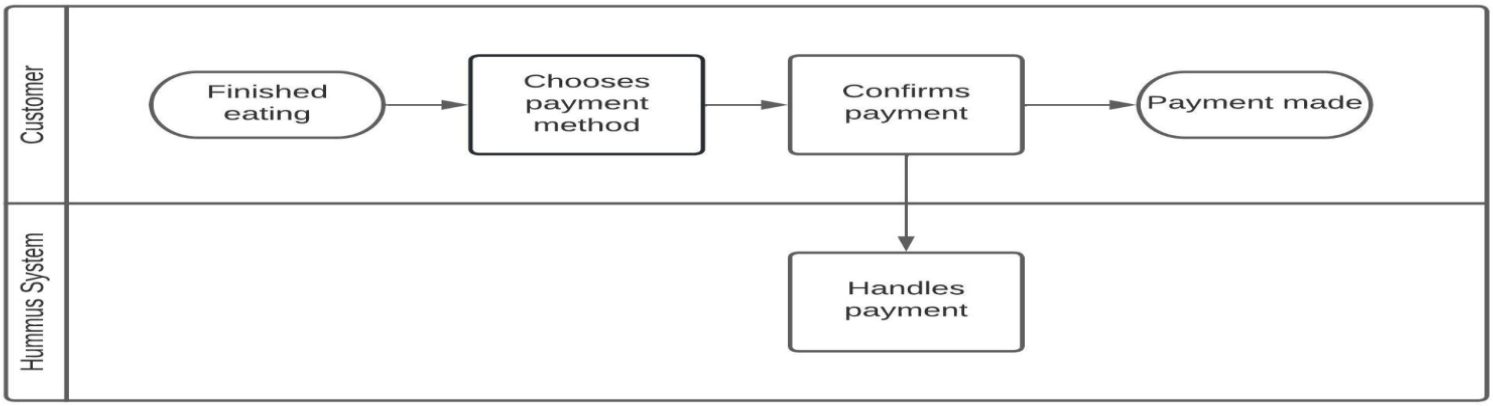


3.Further on in the development, a new task could have been automated and modernized. Payment, as the usual payment method is dependent on the waiter interacting with the customer, the customer depended on the waiter to pay, a solution to this bottle neck could be developed.



4.With Humuus product the costumer can pay directly without being depended on the waiter or cashier.

The customer can complete the payment transaction directly from their table at any time they want immediately.



LO 8- Professionaly – GP + IP:

For the IP research requirements. You can check the hyper link below.

- JWT Research
- Testing Research