

Sprint Ceremony Notes

➤ Roles

- Lev: Product owner
- Chris: SCRUM master
- Kruno: Developer
- Martin: Developer

❖ Sprint X

➤ Sprint planning

- Goal
- Sprint backlog

➤ Daily SCRUM meeting 1

- Completed yesterday
- Plan for tomorrow
- Problems

➤ Daily SCRUM meeting 2

- Completed yesterday
- Plan for tomorrow
- Problems

➤ Daily SCRUM meeting 3

- Completed
- Plan for tomorrow
- Problems

➤ Daily SCRUM meeting 4

- Completed
- For next sprint
- Problems

➤ Sprint review

➤ Sprint retrospective

- Start
- Stop
- Continue

❖ Sprint 1

➤ Sprint planning

- Goal: Rent and place games
 - Small iteration first
- Sprint backlog
 - Broke down task one into sub tasks
 - Took some small simplified versions from other product backlog items

- Daily SCRUM meeting 1
 - Completed yesterday
 - Chris: Repo
 - Kruno: Console commands
 - Lev: Model to add games, barebones login
 - Martin: Game and gamelist
 - Plan for tomorrow
 - Chris: Work on days left counter
 - Kruno: Unit tests
 - Lev: Unit tests
 - Martin: Method for getting all unrented games, client server split (RMI)
 - Problems
 - None
 - Assistance
 - Chris asked Kruno and Lev to take up the unit test task
- Daily SCRUM meeting 2
 - Completed yesterday
 - None
 - Plan for tomorrow
 - Lev and Chris: RMI split
 - Kruno: Javadoc
 - Martin: Adding console commands
 - Problems
 - Limited bookings
- Daily SCRUM meeting 3
 - Completed
 - None
 - Plan for tomorrow
 - Everyone: Fix RMI
 - Everyone: Continue javadoc
 - Problems
- Daily SCRUM meeting 4
 - Completed
 - Martin
 - ◆ Added transactions
 - Kruno
 - ◆ Basic login for RMI
 - ◆ Fleshed out game
 - Chris and Lev
 - ◆ Worked on documentation
 - For next sprint
 - GUI

- Database start
 - ◆ Schema
 - ◆ Diagrams
 - ◆ Tables
 - More javadoc
 - Expand on notes within project report
 - Notes for process report
 - Consider different privileges
 - Problems
 - More pair work
 - Sprint review
 - More use of the product backlog
 - Each day when tasks are being allocated
 - After sprint backlog
 - Demo
 - Return can be completed further
 - Sprint retrospective
 - Start
 - Events on Discord
 - ◆ Helps see the place and time easier
 - Stop
 - Leaving for home during the sprint
 - Being late from lunch
 - ◆ Disrespect of people's time
 - Continue
 - Music
 - Meeting in person
 - Pair work
- ❖ Sprint 2
- Sprint planning
 - Goal
 - GUI
 - Database
 - Sprint backlog
 - Daily SCRUM meeting 1
 - Completed yesterday
 - Everyone: GUI navigation and basic diagram
 - Lev: Started on view model
 - ◆ Extended login functionality
 - Kruno: FXMLs and controllers
 - Martin: Userlist and reading up on connecting database
 - Chris: FXMLs and controllers
 - Plan for tomorrow
 - Chris and Kruno: Continue fxmls and controllers

- Martin and Lev: Viewmodels
 - Add main to run
- Problems
 - Use reformatting more often
 - More clear end to meeting
 - Less work during the meeting
 - Volume: 28 for youtube
 - Less commits
 - ◆ Commit to next semester using branches
 - Write email
- Daily SCRUM meeting 2
 - Completed yesterday
 - Kruno and Chris
 - ◆ Worked on controllers and fxmls
 - Martin
 - ◆ Javadoc, update to User and Game
 - Lev
 - ◆ Viewmodels, admin panel function, login, signup
 - Plan for tomorrow
 - Delete old test classes
 - ◆ Add archive package
 - Server logs everything
 - ◆ No need for old console commands
 - ◆ Class to hold logs
 - Viewmodels
 - ◆ Admin
 - Correct functions
 - Change view name to AddEdit
 - Windows
 - ◆ Every single window must have a way to go back
 - Users
 - ◆ Prevent resizing
 - ◆ Title on top bar of window
 - Alerts
 - Use case work
 - NFR
 - More notes for project report
 - Javadoc
 - Class diagram and domain model
 - Problems
 - Marshalling errors caused by failure to copy classes
 - ◆ Please copy or face the Magyar wrath
- Daily SCRUM meeting 3
 - Completed

- Kruno
 - ◆ Javadoc, return games, user profile
 - Martin
 - ◆ Class diagram, javadoc
 - Lev
 - ◆ Inventory window
 - Chris
 - ◆ Browse view, game info, notes for project report
- Plan for tomorrow
 - Kruno: Use cases + class check
 - Martin: Class diagram, controller instance var to private
 - Lev: continue work on admin
 - Chris: Database - button to add a single thing through the GUI.
Debug controllers
- Problems
 - Merge issues
 - ◆ Resolve manually
- Daily SCRUM meeting 4
 - Completed
 - Kruno
 - ◆ More use case descriptions, return info, class diagram, model methods
 - Lev
 - ◆ Admin panel
 - Martin
 - ◆ Class diagram
 - Chris
 - ◆ Add/Edit functionality
 - For next sprint
 - Kruno
 - ◆ Transactions
 - Lev and Martin
 - ◆ Finish admin panel
 - Chris
 - ◆ Database nonsense
 - Problems
 - Reformat
- Sprint review
 - It was good
 - Doing well on schedule
 - Basic requirements met
 - Database is the only missing core architecture
 - Days missing from rental
 - Review is needed

- Sub
 - Works for 30 days, and then press the pay button
 - Revoke rental priv
 - Balance int
- Sprint retrospective
 - Start
 - More homework
 - Reformat
 - Stop
 - Using phone
 - Bullying Martin
 - ◆ Especially Chris
 - Continue
 - Hard work
- ❖ Sprint 3
 - Sprint planning
 - Goal
 - Transactions
 - Database
 - ◆ Diagrams
 - ◆ Users
 - ◆ Games
 - ◆ Normalization
 - User edit
 - GUI renting
 - Check for date
 - Refund
 - Balance
 - ◆ User side
 - Add to balance
 - Pay for subscription/ fine
 - ◆ Admin side
 - Refund/fine
 - Revoke privileges
 - Ratings
 - Terminate the user
 - Sprint backlog
 - Daily SCRUM meeting 1
 - Completed yesterday
 - Lev
 - ◆ Balance, subscription, transaction, currently logged in user, etc.
 - Kruno

- ◆ More fxmls!
 - ◆ Renting process
 - Martin
 - ◆ Admin panel
 - Edit users
 - Constraints
 - Chris
 - ◆ CRUD methods
 - Plan for tomorrow
 - Lev
 - ◆ Access balance from edit user
 - Kruno
 - ◆ Renting.
 - Martin
 - ◆ Refund, fine, and revoke
 - Chris
 - ◆ Make branch to integrate CRUD methods
 - Problems
- Daily SCRUM meeting 2
 - Completed yesterday
 - Kruno
 - ◆ GR diagram, relational schema
 - Martin and Lev
 - ◆ Admin panel is done
 - ◆ Edited rent and return
 - ◆ Balance
 - ◆ “Everything except the database is done”
 - ◆ Reviews
 - Chris
 - ◆ Implemented CRUD methods for game
 - ◆ Started new branch
 - ◆ Kruno “volunteered” his help
 - Plan for tomorrow
 - Kruno and Chris
 - ◆ Implementing DAO in db test branch
 - Game and users
 - ◆ When you rent a game, the database should reflect it
 - Create a rental and a transaction
 - Lev and Martin
 - ◆ Documentation
 - ◆ Code cleanup
 - ◆ Assisting to modify model methods
 - Problems

- Database has become a critical priority which causes others to wait
- Daily SCRUM meeting 3
 - Completed
 - Kruno
 - ◆ Use cases fleshed out
 - ◆ Diagram updates
 - ◆ Contact button
 - Lev
 - ◆ User database integration
 - ◆ Git conflict resolution
 - Martin
 - ◆ Worked on project description
 - Requirements
 - Bullet point
 - ◆ Javadoc
 - Chris
 - ◆ Game database integration
 - Apply methods to model manager
 - Able to rent and return games
 - See games rented by user
 - Plan for tomorrow
 - Kruno
 - ◆ Class diagram readability
 - ◆ Reports
 - Lev
 - ◆ Javadoc
 - ◆ Code cleanup
 - Martin
 - ◆ Checking for age when renting
 - ◆ Javadoc
 - Chris
 - ◆ Triggers
 - Calculating age
 - Days left
 - Active -> Rented
 - ◆ Reports
 - Problems
- Daily SCRUM meeting 4
 - Completed
 - Lev
 - ◆ Squashing bugs
 - Martin

- ◆ Age check
 - ◆ Javadoc
 - ◆ Test data
 - Kruno
 - ◆ Transactions!
 - Chris
 - ◆ Triggers
- For next sprint
 - Lev
 - ◆ Game info
 - Martin
 - ◆ Testing
 - ◆ Test cases, ZOMB+E
 - ◆ Javadoc
 - Kruno
 - ◆ Documentation
 - ◆ Pair programming and functionality
 - Chris
 - ◆ Documentation
 - ◆ Database veracity
 - ◆ Design documents
- Problems
 - Focus
 - ◆ More five minute breaks
 - Reduce burnout
 - ◆ More social activity
 - ◆ Personal plus feature
 - Something you think is fun
 - ◆ Space out documentation
 - More light caffeine
 - ◆ Tea
- Sprint review
 - Updated the product backlog
- Sprint retrospective
 - Start
 - See most recent problems
 - Define certain amount of daily documentation for each person
 - Social activities when we are ahead of schedule
 - Stop
 - In future semesters, avoid pushing to main branch
 - Leaving trash in the room
 - Contacting steffen when it is not needed
 - Continue
 - Booking rooms well in advance

- Using larger rooms when possible
- ❖ Sprint 4
 - Sprint planning
 - Goal
 - Intense testing
 - Bug squashing
 - ◆ Transaction
 - Documentation
 - Search!!!!
 - Increase error label size
 - Custom error text for expected exceptions
 - Check when error label is reset
 - Sprint backlog
 - Done
 - Daily SCRUM meeting 1
 - Completed yesterday
 - Martin
 - ◆ GameList test cases
 - Lev
 - ◆ Javadoc
 - ◆ Bug fixes
 - ◆ Custom exceptions
 - ◆ Days left in the game
 - ◆ Review
 - Chris
 - ◆ Search
 - ◆ Transactions
 - ◆ JavaDoc
 - Kruno
 - ◆ Transactions
 - ◆ Javadoc
 - Plan for tomorrow
 - Martin
 - ◆ Process report
 - ◆ Test cases
 - Rent, return, browse
 - Lev
 - ◆ Bug fixes
 - ◆ Process report
 - ◆ Update use cases to current version
 - Kruno
 - ◆ Domain model
 - ◆ Class diagram rework
 - ◆ Use case revision

- Chris
 - ◆ Javadoc
 - ◆ Project report: Introduction
- Problems
 - N/A
- Supervisor Meeting 24/5 10 - 11
 - Use cases
 - Which ones to combine to follow BOSS rule
 - Relationship to current solution/ maintaining ambiguity
 - ◆ A: Avoid design elements
 - Relationship to test case steps
 - When should we use activity diagrams instead?
 - ◆ Use system sequence if there is a lot of system interaction
 - ◆ Activity diagram if a lot of steps
 - ◆ Short and precise
 - Process report
 - Size
 - ◆ Similar to 1st semester
 - 1 page
 - How to reflect on supervision
 - ◆ Role of supervisor
 - How did that affect the group?
 - ◆ Management theory
 - ◆ Steffen answers with a question
 - Integrating theory into personal reflections
 - ◆ *Shrug*
 - ◆ Look through learning paths
 - ◆ Use estimate profiles
 - Describe missing
 - ◆ More application instead of explanation of theory
 - ◆ Combine with SCRUM roles
 - Project report
 - How much of the introduction can be lifted from the description?
 - ◆ Sometimes better word for word
 - ◆ Can lead to the solution
 - Diagraming
 - ◆ Class diagram
 - Diagram by package?
 - Improving readability
 - Make class diagram operations and fields
 - Zoom in to show packages
 - Make a note to not show setters and getters

- Use the images of the diagram as the parts come up in the report
 - ◆ Sequence diagram
 - Doesn't feel right
 - Analysis artifact
 - Use both activity and sequence
 - Conventions and astash functionality
 - If and branching
 - ◆ Domain model
 - Inheritance?
 - Not reflected in program, but real world
 - A: "This is fine"
 - Size and detail
 - Avoid design details
 - Show difference when discussing design
 - ◆ Design documents for GUI?
 - User guide contains screenshots
 - Describe for experts how it was designed
 - ◆ Interaction diagrams?
 - Always ask what story you want to tell
- Defining more non-functional requirements
 - ◆ INVEST and FURPS+
 - ◆ Coming from the customer
 - Product owner
 - Avoid the school's project requirements
 - Can be added late
 - Must be justified
 - "Postgres because ..."
- Implementation
 - ◆ Compare to SEP 1
 - ◆ Use special parts of javadoc
 - ◆ How to follow from design
 - Ex: Run later
- Testing
 - ◆ Structuring the section
 - Chronological or v-model?
 - Purpose is to convince the reader that it is working
 - Unit testing
 - Tested for this and this
 - Show fraction of junit test
 - Refer to appendix
 - Show most important test

- Test cases
 - Might be in appendix
 - Not only sunny scenario
 - ◆ Artifacts needed
 - ◆ Acceptance testing
 - Conclusion
 - ◆ Maybe acceptance
 - Comments
 - ◆ For expert audience
 - Consistency on explanations
- Daily SCRUM meeting 2
 - Completed yesterday
 - Supervisor meeting
 - Lev
 - ◆ Javadoc
 - ◆ Exam prep
 - Kruno
 - ◆ Domain model
 - ◆ Class diagram rework
 - Martin
 - ◆ JUnit
 - ◆ Exam prep
 - Chris
 - ◆ Javadoc
 - ◆ Process report: introduction
 - Plan for tomorrow
 - Lev
 - ◆ Process report
 - Kruno
 - ◆ Class diagram rework
 - ◆ Activity/ sequence diagrams
 - Martin
 - ◆ Personal reflection
 - Problems
 - Bring tea
- Daily SCRUM meeting 3
 - Completed
 - Lev
 - ◆ Personal reflection
 - ◆ Process report
 - ◆ Error messages
 - Kruno
 - ◆ Steffen answers to diagram
 - ◆ Start on personal reflection

- Martin
 - ◆ hasRental
 - ◆ Cascade deletion for users
 - ◆ Test cases
 - ◆ Process report review
- Chris
 - ◆ Extend game
 - ◆ Introduction
 - ◆ Analysis
 - ◆ Notes for design
- Plan for tomorrow
 - Lev
 - ◆ More error messages
 - ◆ Limit to extend game
 - Kruno
 - ◆ Standardize use cases
 - ◆ Activity diagrams
 - ◆ Personal reflection
 - ◆ E-estimate profiles
 - GDPR request
 - Martin
 - ◆ More process report
 - ◆ More test cases
 - Renting
 - Terminating users
 - Chris
 - ◆ Scroll fix
 - ◆ Project report notes
- Problems
 - Rooms
 - ◆ We booked as much as we could
 - Health issues
- Daily SCRUM meeting 4
 - Completed
 - Lev
 - ◆ Password security
 - ◆ Limit to extend
 - ◆ More error message
 - Kruno
 - ◆ Standardize use cases
 - ◆ Activity diagrams
 - ◆ Personal reflection
 - ◆ E-estimate profiles

➤ GDPR request

- Martin
 - ◆ Group contract upload
 - ◆ Lev import to personal reflection
 - ◆ Fixed a JUnit test
- Chris
 - ◆ Use case diagram description in analysis
 - ◆ Introduction review
 - ◆ Scroll fix
- For next sprint
 - Lev
 - ◆ Complete JavaDocs
 - ◆ Process report w/ Martin
 - ◆ Pair up to write implementation section
 - ◆ Helping with critical deliverables
 - ◆ Write admin user guide
 - Kruno
 - ◆ Design section in project report
 - ◆ Team up with Lev for implementation
 - ◆ Helping with critical deliverables
 - Link to pdf in program
 - ◆ Write renter user guide
 - Martin
 - ◆ Process report refinement
 - ◆ Test section in project report
 - ◆ Project future
 - ◆ Results and discussion first draft
 - ◆ Help with deliverables
 - Chris
 - ◆ Introduction and analysis refinement
 - ◆ Abstract
 - ◆ Conclusion
 - ◆ Sources
 - ◆ Appendices
- Problems
- Sprint review
 - Passable program for exam
 - Security
 - Business value for stakeholder
 - Real world days to brute force
 - Add easter eggs final sprint
 - Just need link to user guide
 - Print outs on server side
 - Remove game list

- Error label needs reset correctly on user profile
- Transactions xml
 - To be deleted
- Document in the next sprint
- Non-functional
 - Users' passwords are stored securely that cannot be retrieved without brute force.
 - User guide must exist for users and administrators to explain core functions.
 - Database must be able to drop the schema and create the tables again from scratch.
 - The client program must open to the IP address prompt within 60 seconds.
 - The database must be implemented in PostgreSQL so that the current system administrators can use their current knowledge base to maintain it.

➤ Sprint retrospective

- Start
 - Keep attention on documentation
 - Working during the weekend
- Stop
- Continue
 - Meeting in person
 - Watching shit movies

❖ Sprint 5

➤ Sprint planning

- Goal
 - Lev
 - ◆ Complete JavaDocs
 - ◆ Process report w/ Martin
 - ◆ Pair up to write implementation section
 - ◆ Helping with critical deliverables
 - ◆ Write admin user guide
 - Kruno
 - ◆ Design section in project report
 - ◆ Team up with Lev for implementation
 - ◆ Helping with critical deliverables
 - Link to pdf in program
 - ◆ Write renter user guide
 - Martin
 - ◆ Process report refinement
 - ◆ Test section in project report
 - ◆ Project future
 - ◆ Results and discussion first draft

- ◆ Help with deliverables
 - Chris
 - ◆ Introduction and analysis refinement
 - ◆ Abstract
 - ◆ Conclusion
 - ◆ Sources
 - ◆ Appendices
- Sprint backlog
- Daily SCRUM meeting 1
 - Completed yesterday
 - Martin
 - ◆ Personal reflection
 - ◆ Work on execution
 - Kruno
 - ◆ Personal reflection
 - ◆ Design in project report
 - Lev
 - ◆ Fixed transactions (again)
 - ◆ Completed JavaDoc
 - ◆ Security
 - ◆ Started on admin guide
 - Chris
 - ◆ Personal reflection
 - ◆ Analysis expansion
 - Plan for tomorrow
 - Martin
 - ◆ Finish execution
 - ◆ Project future
 - ◆ Results and discussion
 - ◆ Test section
 - ◆ Easter egg
 - Lev
 - ◆ Admin user guide
 - Button
 - JavaDoc for button
 - ◆ Project report implementation
 - Work with Kruno
 - Kruno
 - ◆ Design
 - ◆ User guide
 - Button
 - ◆ Implementation
 - ◆ Personal reflection

- Chris
 - ◆ Conclusion first draft
 - ◆ Abstract first draft
 - ◆ Sources
 - ◆ Look where error label is reset
- Problems
 - Rain
 - Too long part time work
- Daily SCRUM meeting 2
 - Completed yesterday
 - Lev
 - ◆ Project report: implementation
 - ◆ Conclusion refinement
 - ◆ Formatting
 - Kruno
 - ◆ Project report: implementation
 - ◆ Design section
 - ◆ Process report work
 - Martin
 - ◆ Execution
 - ◆ Supervisor
 - ◆ Project future
 - ◆ Test
 - ◆ Process: Conclusion
 - Chris
 - ◆ Abstract
 - ◆ Conclusion: Project report
 - ◆ Process: Introduction
 - ◆ Formatting
 - ◆ Sources
 - ◆ Error label reset checks
 - Plan for tomorrow
 - Lev
 - ◆ Documentation refinement
 - Implementation
 - Design
 - ◆ Moving source code to appendix
 - ◆ Make some questions for Steffen
 - Kruno
 - ◆ Documentation refinement
 - ◆ Look over the results with Martin
 - ◆ Adding to the analysis
 - Martin
 - ◆ Results section

- ◆ Revise implementation
 - Workaround for callback
 - ◆ Easter egg
 - Chris
 - ◆ Triggers in implementation
 - ◆ Review implementation
 - ◆ Review design
 - ◆ Figure out birthday plans
 - Problems
 - Google docs numbered lists
- Supervisor meeting 31/5
 - Implementation within the design section
 - Introduction: Delimit arguments and formatting
 - Breaking up the project future into paragraphs
 - Follow a sprint from start to finish
 - Use the burndown
 - Reflect on culture
 - Specify specs of hardware for test
 - Reflect on e-estimate profiles
 -
- Daily SCRUM meeting 3
 - Completed
 - Lev
 - ◆ Process
 - ◆ Project
 - ◆ Comments
 - Kruno
 - ◆ Intro revision
 - ◆ Analysis punch up
 - Use case
 - Martin
 - ◆ Easter egg
 - ◆ Results section
 - ◆ Process report conclusion
 - Chris
 - ◆ Intro revision
 - ◆ Imported diagrams
 - ◆ Lined up the class diagram
 - Plan for tomorrow
 - Lev
 - ◆ Packaging
 - ◆ Reread both reports
 - Kruno
 - ◆ Astah finishing touches

- ◆ Extract images
 - ◆ Ensure diagrams are in the correct appendix
 - ◆ Reread reports
 - Martin
 - ◆ Test case expansion
 - ◆ Communicate with group 1
 - ◆ Reread reports
 - Chris
 - ◆ Cultural reflection
 - ◆ Proofread exchange
 - ◆ Reread reports
- Problems
 - Documentation burnout
 - Brain rot
- Daily SCRUM meeting 4
 - Completed
 - Lev
 - ◆ Packaging
 - ◆ Reread both reports
 - Kruno
 - ◆ Astah finishing touches
 - ◆ Extract images
 - ◆ Ensure diagrams are in the correct appendix
 - ◆ Reread reports
 - Martin
 - ◆ Test case expansion
 - ◆ Communicate with group 1
 - ◆ Reread reports
 - Chris
 - ◆ Cultural reflection
 - ◆ Proofread exchange
 - ◆ Reread reports
 - For next sprint
 - There is no next sprint!
 - Problems
 - Lack of attention span
 - Severe burn out
- Sprint review
 - It was good
 - Product backlog complete
 - Full functionality
 - Addressed errors
- Sprint retrospective
 - Template not applicable

- Meaningful adjournment
 - Meeting June 1
- Did not work well
 - Hours
 - ◆ Should have started later (9:30 - 10:00)
 - ◆ Others agree with 9:30, but not later
 - ◆ Morning vs. evening person
 - Not documenting methods as written
 - Using #important for the wrong things
 - Lack of read receipts
 - Conversation control
 - ◆ Rooms are not private
 - Particularly with an open
 - Stubbornness
 - ◆ Initial failure to admit wrongs
 - We could have worked more on our weaknesses
 - ◆ We did use our strengths for tasks
 - Database Java connection must be explained in class
 - Starting on half day
 - ◆ Confuses what was completed
 - ◆ Does allow work on weekends, but so does cheating
 - We should have prepared questions for supervisor meetings
 - ◆ We did this toward the end
 - Supervisor meetings could have been documented better
 - ◆ Helped with process report
 - Supervisor challenges
 - ◆ Vague answers
 - “What do you think?”
 - Defining requirements
 - Adapter for table
 - ◆ Indirect critique
 - Must use understanding of facial expression and body language to interpret response
 - Study room condition
 - ◆ Lack of ventilation
 - ◆ Couches taking space on 5th floor
 - ◆ Smell
 - Interruptions in conversation
 - ◆ Yielding more often
 - ◆ Active listening
 - ◆ Eye contact to speaker
 - Phones during project work
 - ◆ Buzz on the table is distracting
 - ◆ Shows disrespect for other people’s time

- Dogpiling
 - ◆ Less care for opinions from victim
 - ◆ Turn speaking
 - ◆ Use in SCRUM
 - Problems in daily
 - Sprint retrospective
- What worked well
 - Meeting in person
 - ◆ Better coworking environment
 - ◆ Shared music
 - Improved punctuality
 - Adaptable breaks
 - ◆ Adapting to individual work flows
 - ◆ Smaller and more numerous
 - ◆ As needed
 - Git
 - ◆ Better version control
 - ◆ Could have improved by using branches
 - Main vs. development
 - Will be expanded upon in DevOps in 4th semester
 - Tea
 - ◆ Shared resource
 - ◆ Light caffeine
 - JavaDoc
 - ◆ Should be written as soon as the method is written
 - Booking the maximum rooms allowed
 - ◆ Using all 8 slots
 - ◆ Compiling it in discord
 - Discord
 - ◆ Channel segregation
 - Completing the work a day ahead of turn in
 - ◆ Reduced stress
 - SCRUM framework
 - ◆ Adapting to stand up meetings
 - Full attention
 - Templates
 - ◆ Ceremonies, backlog, burndown
 - Good rapport with supervisor
 - ◆ Needed to have specific question
 - ◆ Great availability
 - Actually invested in our response
 - Responsive to emails
 - ◆ Very helpful for fine tuning

- Detail oriented
 - ◆ Meaningful comments on description and requirements
 - ◆ Also our teacher for 3 classes
 - Better understanding of dynamics and class lessons
 - Considers feedback on course content to a higher degree
- Reflection
 - ◆ Retrospective
 - ◆ Voicing grievances