

Meeting 18 – Sprint 3 Worksheet

- Attendeed
 - Rainclouded
 - Josh
 - Leeroy
 - Mykola
 - Nathan
- Coming availability
 - Josh Wednesday, thursday + next week
 - Leeroy
 - 1-2h per day
 - Rainclouded
 - Probably not until after
 - Nathan
 - Available until tuesday
 - Mykola
 - Available after today
- Goals
 - Code freeze on friday
 - Merge demo to dev
 - Bug hunt till then
 - Start the security & load test
 - Update & validate documentation
 - Deploy
 - Pipeline organization cleaning
 - Known bugs
 - Sign in permissions
 - Add/remove account to permission group does not update the server
- Commitmets
 - Josh – Deployment – fix go tests
 - Leeroy – Load testing & fixing tests
 - Mykola – bugs & cypress tests
 - Nathan – Cypress tests
 - Rainclouded – Security analysis
 - Is flask okay

-
- Thoughts
 - Offload to third party
 - Learning
 - Saving dev time
 - Maybe auth/login
-
- Other Thoughts
 - What do diff
 - Not use flask (security)
 -

Meeting 17 – Sprint 3 & Demo

Nov 26

Attendees

- Nathan
- Rainclouded
- Mykola
- Leeroy
- Josh

Meeting objective

- Prep for demo
- Discuss remaining tasks for sprint 1

Notes

- Remaining Tasks
 - Focus (up to presentation)
 - Get UI presentable
 - Rainclouded
 - Formatting
 - Account integration
 - Josh + Mykola
 - Accounts integration w/ permissions
 - Leeroy

- Populate room number on login
- Nathan
 - UI refactor mostly complete
- Presentation
 - Prerecorded demo
 - Each feature is its own video
 - Presenters
 - Leeroy
 - Presentation Ideas
 - Lots of code – so much code
 - # of endpoints
 - Distribution
 - Analysis: how much code is “unique” to the service
 - Eg, setup/config/DTO is generic
 - Business logic is unique
 - Variety of tech stack
 - Rainclouded – general issues (coordinating & testing)
 - Consistency
 - Code switching & environment
 - Solving the same problem
 - Human Features (anonymous)
 - How we balance our negative and positives
 - Weakness
 - Overcommitting
 - Problem: Stalled on WIP
 - Solution: retrospective reorganized priorities and responsibilities
 - Lack of experience
 - Problem: didn’t understand discussion at beginning
 - Solution: Took on more work
 - Better solution: Go through explanations slower and provide more support
 - Focused on little details
 - Problem: Costed time, stalling other important tasks
 - Solution: Reevaluate task priority

- Go into detail later
- Yapping
 - Issue: Distracted, longer meetings
 - Solution: locking
- Low communication
 - Problem: Group didn't know where WIP was at
 - Solution: Keeping track of communication
- Positives
 - Consistent high-volume work
 - Really solid details
 - Expertise
 - Maintaining positive atmosphere
 - Filling the missing skills
 - Leeroy scheduling the room
- Individual take on microservices
 - What did you think?
 - Good
 - They are one of the architectures
 - Very cool that it's decoupled
 - No merge conflicts
 - Nice that you didn't have to worry about other developers code
 - Bad
 - Overhead
 - No clear understanding of others codebases
 - Challenges on code review
 - Different languages is a challenge
 - Eg, urls with trailing slash was handled differently between languages
 - Overwhelming
 - Lot to learn
 - Lot to go through
 - When something goes wrong
 - Infrastructure issue where?
 - Leeroy's container volume story
 - Mykola's trailing slash story
- Where the project could go

- Extensions
 - Account customization
- Prototyping vs result
 - Challenges & notes
 - Transition challenges
 - Highlighting design
- No Third part APIs
 - Was that a good choice?
 - Good
 - Good for cost
 - More flexible to modification
 - No reliant on other software
 - Bad
 - Costs time –
 - couldn't get everything that we wanted
 - Accounts are slim, trimmable features missed
 - Missed experience
- Managing expectations
 - Meeting 1 was a surprise,
 - So we had a checkin during sprint two to align expectations
- Jobs
 - Screen recording
 - Leeroy
 - Setup presentation
 - Rainclouded
 - Design topic
 - Nathan
 - 2 demo videos
 - Leeroy
 - Leeroy checks out

Meeting 16 – Technic pres Practice

Attendees

- Rainclouded
- Leeroy

- Josh

Presentation

- Intro – 1:40
- Benefits – 3:14
- Complications – 5:11
 - Remove encrypted traffic
 - Maybe remove zero trust
 - Consolidate deployment&Testing
- Conclusion – 0:40

Presentation Checklist

- Speak loudly
-

Meeting 15 – Technique presentation Practice

Attendees

- Leeroy
- Josh

Meeting Objectives

- Practice and refine the technique presentation.
- Clarify key points about microservices and their benefits.
- Address gaps in the presentation

Presentation Breakdown:

Intro to Microservices (1m 19s):

- **Topics:**
 - Microservice Architecture.
 - Comparison to Monolithic systems (COMP 3350 context).
 - Emphasize motivation and clarity.

Presentation Content:

1. Leeroy (3:40)

- **Overview:** What is a microservice?
- **Key Discussion Points:**
 - Risks to the system.
 - Learning and experimentation opportunities (languages, DBs, technologies).
 - Importance of isolating bad or messy code to reduce impact.
 - Career paths: Remove "given a choice" phrasing for clarity.
 - Finding bugs: Highlight issues related to miscommunication.

2. Josh (6:40)

- **Key Points:**
 - Maintenance challenges: Managing 51 endpoints.
 - Prioritization: Reduce unnecessary complexity.
 - API traffic: Example of exponential increase from additional messages.
 - List Solutions
 - Delivery issues: Practice to reduce filler words ("Ums" and "Ahs").

Meeting 14 – Technique sharing presentation

Nov 19

Members

- Josh
- Rainclouded
- Nathan
- Mykola

Meeting objective

- Brainstorm Presentation points
- Assign tasks & roles

Possible Points

- Coordinating API & Client interface/concrete classes
- An API that has more exposure (eg, internal endpoints, like session/me)
- Reverse Proxies
- Specialization

Responsibilities

- Presenters
 - Josh – wants to
 - Rainclouded – wants to
 - Leeroy – wants to
 - Mykola and Nathan will do an interpretive dance in the background (joke)
(review and edit the presentation)
 - Both willing to take on portions of the presentation if a presenter is feeling overwhelmed

Meeting 13 – Sprint 3 Responsibilities

Nov 15

Members

- Rainclouded
- Leeroy
- Josh
- Mykola
- Nathan

Meeting objectives

- Goals for Sprint 3

Missing Elements

- ☐ ~~Frontend Create Staff Account~~
 - Mykola
- ☐ ~~Frontend Create Guest Account~~
 - Mykola
- ☐ ~~Permission-account connection~~
 - Josh+Rainclouded
- ☐ ~~Edit account details~~
 - Mykola
- ☐ ~~Account-Permission group~~
- ☐ ~~Guest-View their tasks status~~
 - ☐ ~~Backend~~
 - Leeroy

- ☐ ~~Frontend~~
 - Nathan
- ☐ ~~Test Non-functional requirements~~
 - Nathan
- ☐ ~~Coverage reports~~
 - Leeroy
- ☐ ~~Hide buttons if not permitted~~
 - Nathan
- ☐ ~~Automated deployment~~
 - Josh
- ☐ Tools Pentation (Nov 25) - Plan on next meeting
- ☐ Final presentation (Nov 29) -- Plan on next meeting

Goals for Sprint 3

- Rainclouded
 - Maintain Coding standards + comments&naming
 - Accounts—Permissions W/ Josh
- Josh
 - Permissions—Accounts W/ Rainclouded
 - Automated Deployment
- Leeroy
 - (backend) Guest – View their tasks status
 - coverage reports
 - Refine Guest frontend (amenities + incident report pages)
- Mykola
 - Create Staff Account
 - Create Guest account
 - Edit Account Details
- Nathan
 - Non-functional (JMeter)
 - Guest View tasks
 - Hide non-permitted buttons
- Deadline:
 - Code completed by Nov 25

Meeting 12 – Sprint 2 Worksheet

Nov 5

Members

- Leeroy
- Rainclouded
- Koala
- Josh

Meeting Objectives

- Sprint success analysis
- Identify Responsibilities for the Worksheet
- Identify missing materials for the worksheet

Worksheet responsibilities

- Rainclouded
 - Testing question
 - Testing plans
- “Not Tested” question
 - Everyone responsible for their own systems
 - Josh – Permission & amenities
 - Leeroy -
 - Koala -
 - Nathan -
 - Rainclouded -
- Josh
 - Profiler
 - Meeting notes refactor/format
- Koala
 - Last Dash
 - Favourite code
- Leeroy
 - Demo works
 - Integration
 - Favourite code

Worksheet work

- Forseen issues
 - UI -> refactoring without breaking systems

- Golang service -> only 1 member can work on it
- Integrating watchman into our CD pipeline
- Load balancer integration
- Wrangle scope creep & harden system
- Documentation
 - Each service needs:
 - Endpoint openAPI description
 - Sequence Diagram
 - README
 - Env Vars
 - Endpoints
 - How to run
 - How to run tests
 - Architecture diagram
 - Project overview architecture diagram
 - Project overview README
- Code Quality: Lint with ctrl+shift+i
 - Josh – Add comments to permissions
 - Rainclouded – Accounts
 - Leeroy – Amenities & Incident Reports
 - Koala – Task system UI
 - *Please claim a service to code quality review (eg, add comments)*

Conclusion:

- All members of the team are now caught up with the project and know their tasks as we prepare for the grading of Sprint 2

Meeting 11 – Sprint 2 Coding Wrap up

October 22

Members

- Josh
- Nathan
- Rainclouded
- Koala
- Leeroy

Meeting Objectives

- Progress check-in: How is it going
- Reevaluate Friday Code deadline
- Check objectives

Progress Check in

- Josh
 - Finished Amenities DB & Tests
 - Permissions – Login
 - Expect Permission group management will be done by Friday
- Rainclouded
 - Env Var started
 - Accounts DB setup and work (no tests)
 - Squashed sprint board bugs
 - Has availability to take on more work
- Koala
 - Focused on frontend for the Task System

The screenshot displays a 'Task Manager' interface. At the top, there's a 'Task Manager' header with a 'Filter' button. Below it is a table with columns: 'Room No.', 'Type of Service', 'Description', 'Time Created', 'Assignee', and 'Status'. The table lists several tasks, including 'Room Cleaning', 'Food Service', and 'Room Cleaning'. Below the table, there are two 'Task Details' pop-up windows. The first window shows details for Room Number: 102, Type of Service: Room Cleaning, and Status: Pending. The second window shows details for Room Number: 104, Type of Service: Food Service, and Status: In Progress. Both windows have buttons for 'Assign to Me' and 'Mark Complete'.

Room No.	Type of Service	Description	Time Created	Assignee	Status
101	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
102	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
103	Food Service	Dinner requested to be served at 7 PM in the room.	10/21/24 11:00 AM	Jane Smith	In Progress
104	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
105	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
106	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
107	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
108	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
109	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
110	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
111	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
112	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
113	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
114	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
115	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
116	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
117	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
118	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
119	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending
120	Room Cleaning	Guest requested quick clean of the entire room including dusting, vacuuming the carpets, wiping down mirrors and windows, and changing the bed linens. Additionally, the guest asked for the trash to be emptied and the bathroom to be thoroughly sanitized. They also mentioned needing more toiletries, including shampoo, conditioner, and body lotion.	10/21/24 8:00 AM	Unassigned	Pending

- Working on integration and additional features
- Friday deadline: functionally doable
 - Tests need writing might be more demanding

- Leeroy
 - Task System feature – Guest UI
 - Incident Reports DB
 - Done, waiting on PR
 - Working on
 - Task system UI
 - UX fix
 - DB
- Nathan
 - Task backend Looking good
 - Validation/tests need to be done

Code Deadline evaluation

- Preference for Saturday deadline

Incomplete / missing components

- Release Tags
- Github Issues: Proper labels & time estimates
- API doc update
- Bug Hunting
- Sufficient comments

Critical objectives

- CD
 - Josh + Rainclouded (next week deadline)
- Tests (Mostly integration and e2e)
 - Rainclouded working on on Saturday

Conclusion:

- All members of the team are now caught up with the project and know their tasks as we prepare for the grading of Sprint 2

Meeting 10 – Task System Feature Planning

Members

- Koala
- Nathan

- Leeroy

Objective

- Plan out a vision for the Task System Feature,
- how the guests will request services and how staff will fulfill them. Create endpoints and object classes

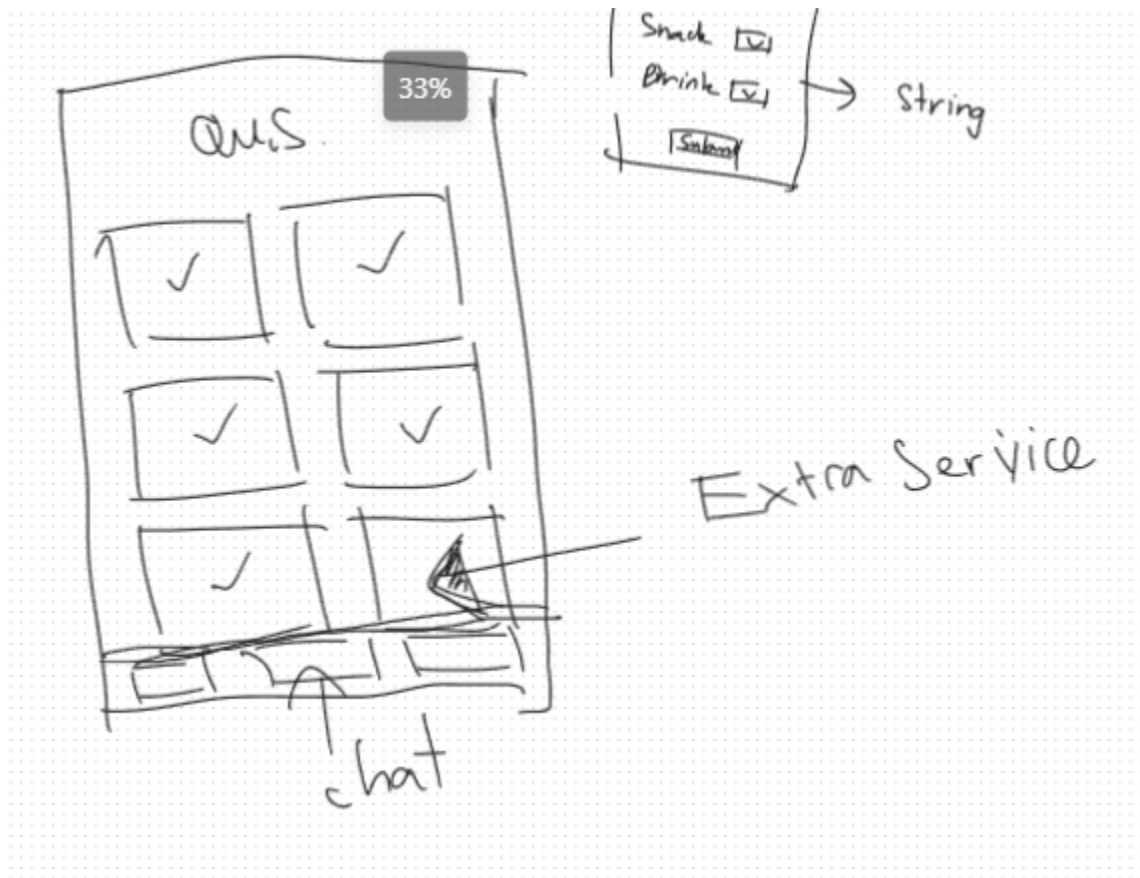
Flow

- Necessary User Stories:
 - Guest creates task
 - Staff claims a task
 - Staff marks task as complete

Types of Requests

- Room Cleaning
 - Text: special instructions
- Maintenance
 - Text: special instructions
- Food Delivery
 - Custom dropbox for what kind of food you want
- Wake up call
 - Time: special instructions
- Laundry Service
 - Text: special instructions
- Spa & Massage
 - Text: special instructions

Guest Front-End



- Quick services displayed on home page
- Other features showed on footer

Staff Front-end

- Table approach
- Priority is inferred using creation timestamp
- Filter options

Endpoints

Localhost:8081/tasks/

- GET tasks
 - Filter option using parameterized urls
 - Status
 - Creation date
 - Assignee

- Task type
- Get task by ID
- Update task
- Create task

Variables

- Title
 - Type of service
- Description
- Room id
- Task id
- Requester account id
- Staff account id
- Status
- Creation timestamp

Conclusion

- All members now have one shared vision and are now prepared to fulfill this feature

Meeting 9 – Sprint 2 contributions commitments

- Meeting notes template:
 - Title / goal
 - Members attendend
 - Objective
 - Conclusion
 - Misc Notes
- Sprint commitments
 - Leeroy
 - Incident reports DB
 - Task guest UI
 - UX fix
 - Task DB
 - Josh
 - Login&Accounts integration
 - UI

- [in review] Amenities DB
 - Permission DB
 - Login&Accounts integration
- Rainclouded
 - Backup
 - CI/CD pipeline
 - Bug squasher
 - Env Var endpoints
- Koala
 - Accounts DB
 - Task Staff UI
- Nathan
 - Task Server
- Bug fix approach
 - Free for all
 - Assign to Rainclouded
 - Message in chat

Meeting 8 – Sprint 2 Proposed contributions

- Review of sprint 1
 - Leeroy
 - Be prepared before the due date
 - Have a dedicated review period
 - Cramming 2 days before
 - Should have “crammed” the weekend before, so the last days were review rather than triage
 - Nathan
 - Issue: Slowed down in the middle
 - Eg, pull requests bottleneck
 - Meeting improvements
 - Start with updates (what you have been working on)
 - Group Chat
 - Casual updates in the group chat initiate, finish, or have trouble
 - Koala
 - Improving PRs

- Include description of what has been changes
 - Pictures, expected behaviour
 - le, a recap
 - Josh
 - PR review minimal
 - Run code
 - Run relevent tests
 - Test what you think it should do
 - (postman)
 - Read code at least once
- Sprint 2 Plan
 - Leeroy
 - Incident Reports Database Implementation (MongoDB, Postgres)
 - Refine Incident Reports Staff UI
 - Task System Staff UI Implementation
 - Nathan
 -
 - Koala
 -
 - Josh
 - Login
 - network integration
- Business analytic
 - Incidents
 - Recent incidents
 - Unresolved & old incidence
 - Aggregate view of all (eg, pie chart of status)
 - Error requests + logging
 -
- TODO
 - Breakdown task system
- TA Meeting:
 - Grade Sprint 1
 - Proposed vs actual contribution
 - Show 1/3 complete
 - Show (prepare) features/User stories/tasks assigned complete
 - PRs reviewed

- Prove DBs were 90% complete
 - Expectations for sprint 2
 - Ours
 - Everyone does full stack x1
 - Pull request reviewers are the responsible party
- Changes
 - Server Healthcheck endpoints
 - Interface, mock, live clients
- Goals
 - UI Endpoint uses env vars
 - CD (GitHub actions)
 - 1 Prod
 - 24/7
 - Updates on merge to main
 - 1 Dev
 - 24/7
 - Updates on merge to dev
 - K8s? Still haven't figured out how to do on azure for free
 - Estimate \$0-\$80
 - Timeline:
 - [Oct 25] - All features complete and in dev
 - New Features
 - Live update incident reports (staff)
 - Login proper
- Possible features
 - Tasks
 - Login
 - Centralized logging?
 - A server + dashboard for staff?
 - All services send information:
 - Requests received/sent
 - Errors
 -

Meeting 7

Oct 8th

- Priorities for this week & last minute work
 - [high] Get everything in dev
 - ☐ Code review
 - [low] Build sessions, permissions, integrate
 - ☐ With Amenities & Reports
 - [high] [sequence diagram](#)
 - ☐ Everyone does their services
 - ☐ Josh compiles into big
 - ☐ Make teams post for each service
 - [high] architecture diagram
 - ☐ From a system POV & for each service
 - ☐ Same as Sequence (josh compiles)
 - [high] create a subset api document (for currently implemented features)
 - ☐ Josh uses readmes to build that
 - Test on postman
 - [high] functionality only indicated for accounts server
 - ☐ Sprint 1 docs directy
 - [high] tests CICD pipeline
 - [medium] worksheet
 - ☐ Mention Cypress in worksheet
 - ☐ Mention minimal seams
 - ☐ Coverage reports
 - Leeroy for .net coverage reports
 - ☐ Most important tests
 - ☐ Add to worksheet document on teams
 - ☐ Acceptance + Integration : cypress
 - ☐ Reproduce team 7 environment
 - Rainclouded + Josh

Meeting 6

Oct 1st, 2024

- TODO
 - Sequence diagrams for all core features
 - Once features are decided, tag related issues on repo
 - Set up CI pipeline
 - Auto build

- Code review
 - Regression test
- Testing plan
- Commitments
 - Rainclouded
 - Authentication
 - Tests
 - CI Pipeline
 - Nathan
 - Routing for login
 - React
 - angular
 - Simple login and home page front end
 - Leeroy
 - Finish incident reports
 - Sequence diagrams
 - Koala
 - Simple UI for Guest
 - Josh
 - Review pull request

Meeting 5

Sept 28, 2024

- TODO
 - Update KANBAN –Leeroy
 - Update Worksheet - Koala
 - Create new Diagram - Josh
 - Update main README - Rainclouded
 - Include project tree
 - Work on foundation implementation
 - Example frontend (amenities) – Nathan
 - Example (amenities) - Leeroy
 - Example DB (Tasks) - Josh
- Priorities
 - Redesign core features + decide on focus for iteration 1
 - Enable stakeholder to decide focus features for sprint
 - Update Worksheet to reflect that

- Diagram
 - Core features
 - Canaban
- Update front facing readme
- Prototype implementation of everything
- Documents updates
 - Create readme with links
 - Including gradable material
 - Including directory tree + links to readmes
 - One line describing server objective
- Task heirarchy
 - Feature: [name] + label
 - includes checkboxes for US
 - US: amenities/[name] +Lebels
 - Includes checkbox for tasks
 - Dt: [name]
- Features
 - **Feature:** Task system (split?)
 - User side
 - Create tasks
 - **User Story:** As a hotel guest, I want to create service requests (e.g. room service, extra towels) so that my needs are promptly addressed
 - **Acceptance Criteria:**
 - **Scenario:** Guest creates a service request.
 - Given I'm a guest, when I open the guest task dashboard and choose to create a new service request, I can enter the request details (type of service, description, priority).
 - When I submit the request, it appears in my task list with a status of "pending," and the relevant staff members are notified.
 - View task status
 - **User Story:** As a hotel guest, I want to view the status of my service requests so I can know when my needs will be met.

- **Acceptance Criteria:**
 - **Scenario:** Guest views the status of service requests.
 - Given I'm a guest, when I open the guest task dashboard, I can see a list of my active requests with their status ("pending," "in progress," "completed").
- Staff side
 - View incomplete tasks
 - **User Story:** As a hotel staff member, I want to see a list of all incomplete guest requests so I can prioritize my work
 - **Acceptance Criteria:**
 - **Scenario:** Staff views a list of incomplete tasks.
 - Given I'm a staff member, when I open the staff task dashboard, I can see a list of all incomplete tasks with details such as request type, room number, and priority.
 - Claim a task
 - **User Story:** As a hotel staff member, I want to claim a task so that I can take responsibility for completing it
 - **Acceptance Criteria:**
 - **Scenario:** Staff claims a task.
 - Given I'm a staff member, when I open the staff task dashboard and click to claim a task, I am assigned as the responsible staff member, and the task status changes to "in progress."
 - Mark task complete
 - **User Story:** As a hotel staff member, I want to mark a task as complete once it's finished so that I can keep my work queue up to date
 - **Acceptance Criteria:**
 - **Scenario:** Staff Member completes a task
 - Given I'm a staff member, when I am on the tasks staff dashboard, when I am viewing an existing task that I have just completed, I can click the "Completed" button and the system updates the task in real time and gives a

confirmation alert.

- **Feature:** Accounts
 - Login (staff & guest)
 - **User Story:** As a user I need to be able to access the tools relevant to me so that I can make use of the system.
 - **Acceptance Criteria:** Users can log in with credentials and are directed to role-specific dashboards (guest, staff, manager).
 - Creation (staff accounts & room accounts)
 - **User Story:** As a hotel manager, I need to be able to give access to guests and employees so they can do their work
 - **Acceptance Criteria:** Hotel manager should be able to create and delete all staff and user accounts
 - **Scenario:** Creating a new account
 - Given I'm a logged in staff with permission to create staff accounts, when I fill in the necessary fields for user creation and submit it, a new account is created.
 - Permissions
 - **User Story:** As a manager, I need to be able to restrict access to various tools so that my services remain secure
 - **Acceptance Criteria:** Managers can assign or modify permissions for different user roles, controlling access to sensitive features.
 - **Scenario:** Editing permissions
 - Given I'm a logged in staff with permission to edit permissions, when I create a new group, configure the available permissions for that group, and assign accounts to that permission group, those accounts gain access to those permissions.
 - Settings
 - **User Story:** As a user. I want to be able to tailor the tools to my needs so that I can engage with the way that I want.
 - **Acceptance Criteria:** Users can update settings (e.g. default dashboard views, notification preferences, and initial landing pages)
 - **Scenario:** Modifying preferences

- Given I'm a logged in staff with permission to edit permissions, when I create a new group, configure the available permissions for that group, and assign accounts to that permission group, those accounts gain access to those permissions.
 -
- **Feature: Amenities**
 - View
 - **User Story:** As a guest, I want to access hotel amenity information so I can find out when amenities are available (eg, the hotel swimming pool is open)
 - **Acceptance Criteria:**
 - **Scenario:** Guest views hotel amenity information
 - Given I'm a guest, when I open the amenities guest dashboard, I can see a list of amenities offered by the hotel. Each amenity provides their operating hours, description and title.
 - Create, update, delete
 - **User Story:** As a staff member, I want to either create, delete or update hotel amenities so guests are always up to date with our amenity information.
 - **Acceptance Criteria:**
 - **Scenario:** Staff updates an amenity
 - Given I'm a staff member, when I open the staff amenities dashboard, and I am viewing an existing amenity, when I update the amenity details and click "Save", the system updates the report in real-time and shows a confirmation message.
- **Feature: Incident reports**
 - View
 - **User Story:** As a hotel manager, I want a way to view past and current incident reports created by my staff to better monitor the hotel incidents
 - **Acceptance Criteria:**
 - **Scenario:** Hotel manager views past and current incident reports

- Given I'm a hotel manager when I access the incident reports dashboard, then I can see a list of past and current incident reports created by my staff. Each report then includes details such as severity, status, and description.
- Update
 - **User Story:** As a hotel manager, I want a way to update incident report details its relating case progresses in real time
 - **Acceptance Criteria:**
 - **Scenario:** Hotel Manager updates an incident report
 - Given I am a hotel manager, and I am viewing an existing incident report, when I update the incident details and click "Save", the system updates the report in real-time and shows a confirmation message.
- **Feature:** Business Analytics (TRIMMABLE)
 - No Guest
 - Staff
 - User Story: As a staff member, I want a way to view a guest's charges incurred in their stay so I could better review and verify their charges
 - Scenario: Given I'm a staff member, when I access the guest billing dashboard, I can select a specific guest's reservation info.
- **Feature:** Booking service (TRIMMABLE)
 - Guests
 - See available dates & rooms & prices
 - As a guest, I want a way to check room availability so I can plan my next hotel stay
 - As a guest, I want a way to view pricing details for hotel rooms so I could budget my stay
 - Staff
 - Create Room Rules
 - Manage reservation

Meeting 4

- Web application distribution server choice

- Development port convention?
 - Currently developer's discretion
 - Note, these ports are only for internal development and testing.
 - Staff: 4200
 - Guest: 5173
 - Accounts: 50001
 - Permissions: 50002
 - Sessions: 50003
- Comment coverage vs self-documenting
 - Examples:
 - Dockerfile

Meeting 3

- TA meeting
 - What to do still
 - When available
 - Thursday: 10:30
 - Friday: 12:30, 1:30
- Sprint goals
 - Create a “standards” doc within the repo (include naming conventions, branching strat, code review expectations)
 - Setting up azure environment
 - Josh
 - Auth
 - Initially start with many permission requests
 - Permissions as a key will be implemented, but not necessarily used
- Task distribution & assignment\
 - Nathan + Koala
 - Build front end skeleton
 - Leeroy + Rainclouded
 - Build the services
 - Josh
 - Build DB example
 - Build entire Tasks service

Meeting 2 Agenda

- Demo presentation (Timed)
- Public-ify repo
- Next steps
- CI/CD pipeline
 - Linting
 - Python – flake8
 - Js – ESLint
 - C# - sonarLint?
 - Webhook – teams
 - Automate Tests
 - Based on changed code section
- When to write code:
 - Josh post demo into repo
 - Rainclouded will update server code
 - Nathan / Koala will mock up ui
 - Leeroy -> webhook
- Naming strats:
 - Git: branch_type/feature/issue_# (all lowercase)
 - feature/account_server/88989
 - Possible linters:
 - Js/ts prettier
 - Js/ts eslint
 - Python – flake8
 - Python – black
 - C# - sonarLint
 - Style guide:
 - Python <https://google.github.io/styleguide/pyguide.html>
 - Python <https://peps.python.org/pep-0008/>
 - C# - <https://google.github.io/styleguide/csharp-style.html>
 - Js/Ts <https://google.github.io/styleguide/jsguide.html>
 -

Meeting 1 Agenda

Sept 7, 10:00 AM

- finalize on name
 - Concierge
- Decide on tech stack (for presentation)
 -
- Core features
 - Login & auth
 - User Story: As a manager, I want to maintain profiles for my employees and guest so they can use appropriate services.
 -
 - Guest services
 - User Story: As a guest, I want I want to easily utilize hotel accommodation to improve my stay
 - Staff dashboard
 - User Story: As staff, I desire to view and respond to guest requests, so they have a better experience
 - User dashboard
 - User Story: As a user, I want to see information about my stay to help me stay organized
 - Non-functional
 - User Story: As a hotel owner, I want to support 500 users making a request over second for efficient access during peak times.
- Dev preferences
 - Leeroy
 - Backend
 - Rainclouded
 - DB
 - Github
 - Koala
 - Frontend
 - Nathan
 - Frontend
 - Josh
 - Backend
 - DevOps

Meeting 1 Agenda

Sept 7, 10:00 AM

- Lock-In Project Idea
 - Rainclouded's Hotel description:
 - Wakeup call, amenity info
 - Based on room (not person)
 - Staff have their own access
 - Reservation
 - Staff directory
 - Reports
 - !Locked in!
- Contributions & Expectations
 - Regular Meeting
 - Petition 2 meetings
 - Standup on Tuesday
 - Distributing Work
 - Github Kanban
 - Sprint 0: jira & **prod manager** of each peice
 - Incomplete Work Protocol
 - Communication Expectations
 - Comminication
 - Entirely through teams
- Version Control
 - Branching Strats
 - Code Review
 - Coding Standards
 - Task Board
 - Github kanban
- Architecture & Tools
 - Architecture:
 - Tentative Microservice:
 - TODO try getting the demo project
 - Tech Stack Options
 - Hosting Service
 - CICD Solutions
 - Documentation + Tests
 - Features
 - Spreadsheet to ideate
- Josh's Microservice Demo

- Assign Team Leads

TODO:bv