|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Category | Requirement | Priority (1 highest) | Source | Notes | feedback |
| App-Com-1 | Application/Communication | Can retrieve list of lifeforms | 1 | Brainstorm |  | some specifications are to wide |
| App-Com-2 | Application/Communication | Can retrieve list of messages | 1 | Brainstorm |  | other are to narrow and can be groupped togehter with outers |
| App-Tech-1 | Application/Tech | When posting a message the app should auto add time | 1 | Brainstorm |  | E:g. "Login Page" is very wide and should explain what should be done (see new one) |
| App-Tech-2 | Application/Tech | When posting a message the app should auto add location | 1 | Brainstorm |  | others like read and write data from context are very specific and could be groupped together like "Read/Write from context" |
| App-Tech-3 | Application/Tech | When posing a message the app should auto add user id | 1 | Test Cases |  | also read what? from the context |
| App-UI-1 | Application/UI | Feed page displaying all mesages | 1 | Brainstorm |  |  |
| App-UI-2 | Application/UI | Page for posting a message | 1 | Brainstorm |  | should be small and precise enough to be be testable |
| App-UI-5 | Application/UI | All pages are responsive | 1 | Brainstorm |  | take a look the test cases for ideas (e.g. adding a letter after |
| Context-1 | Data Storage | Can read and insert data to and from context via backend | 1 | Brainstorm |  |  |
| CQRS-1-1 | CQRS | Backend can handle commands | 1 | Brainstorm |  |  |
| End-2 | Endpoint | Endpoint for getting all messages | 1 | Brainstorm |  | all services entries corresponds to a handler with call to the context |
| End-3 | Endpoint | Endpoint for getting single message | 1 | Brainstorm |  |  |
| End-4 | Endpoint | Endpoint for getting all lifeforms | 1 | Brainstorm |  |  |
| End-6 | Endpoint | Endpoint for posting a message | 1 | Brainstorm |  |  |
| End-7 | Endpoint | Endpoint for getting single lifeform | 1 | Brainstorm |  |  |
| Error-1 | Error Handling | Binary flag from validation can be converted to error messages | 1 | Brainstorm |  |  |
| Factory-1 | Factory | Factories to create entities, one for each aggregate root | 1 | Brainstorm |  |  |
| Factory-1a | Factory | Factory for User | 1 | Brainstorm |  |  |
| Factory-1b | Factory | Factory for Message | 1 | Brainstorm |  |  |
| Factory-1c | Factory | Factory for Lifeform | 1 | Brainstorm |  |  |
| Factory-5 | Factory | Factories uses Result Pattern | 1 | Brainstorm |  | not really a requirement, it might be better formulated as 'Factories can return entity or errors if validation fails' to make it a requirement |
| Repo-4 | Repository | Repository Layer uses transformation mapping | 1 | Brainstorm |  |  |
| Repo-5 | Repository | Repositories can be queried | 1 | Brainstorm | for CQRS |  |
| Repo-6 | Repository | Repository saving should be done via a unit of work | 1 | Brainstorm | ensures all repositories are saved at the same time |  |
| Repo-7 | Repository | Repository for each aggregate root | 1 | Brainstorm | domain driven design |  |
| Repo-8 | Repository | Base repository used by the abstraction repositories | 1 | Brainstorm |  |  |
| Result-1 | ResultPattern | Result Pattern implemented | 1 | Brainstorm | needed to handle errors in a graceful way |  |
| Result-2 | ResultPattern | A result can be converted to a http type (GET, DELETE, POST etc.) | 1 | Brainstorm | extension method |  |
| Security-2 | Security | Hash and salt user password before storing it | 1 | Brainstorm | needed as the user model shall not depend on entityframework core |  |
| Service-1 | Service | Service for User | 1 | Brainstorm | two groups, a command folder and a query folder |  |
| Service-2 | Service | Service for Message | 1 | Brainstorm | two groups, a command folder and a query folder |  |
| Service-3 | Service | Service for Lifeform | 1 | Brainstorm | two groups, a command folder and a query folder |  |
| Service-3a | Service | Can add animal | 1 | Brainstorm |  |  |
| Service-3b | Service | Can delete animal | 1 | Brainstorm |  |  |
| Service-3c | Service | Can update animal | 1 | Programming |  |  |
| Service-3d | Service | Can add plant | 1 | Brainstorm |  |  |
| Service-3e | Service | Can delete plant | 1 | Brainstorm |  |  |
| Service-3f | Service | Can update plant | 1 | Programming |  |  |
| Service-3g | Service | Can pull all plants | 1 | Brainstorm |  |  |
| Service-3h | Service | Can pull all animals | 1 | Brainstorm |  |  |
| Service-3i | Service | Can pull details about one plant | 1 | Brainstorm |  |  |
| Service-3j | Service | Can pull details about one animal | 1 | Brainstorm |  |  |
| Service-4 | Service | Service method for each endpoint | 1 | Brainstorm | endpoints as lean as possible |  |
| SpecPtn-1 | CQRS | Specification Pattern for data validation | 1 | Brainstorm |  |  |
| SpecPtn-1a | SpecificationPattern | Can validate animal creation request | 1 | Brainstorm |  |  |
| SpecPtn-1b | SpecificationPattern | Can validate plant creation request | 1 | Brainstorm |  |  |
| SpecPtn-1c | SpecificationPattern | Can validate animal update request | 1 | Brainstorm |  |  |
| SpecPtn-1d | SpecificationPattern | Can validate plant update request | 1 | Brainstorm |  |  |
| SpecPtn-1e | SpecificationPattern | Can validate user creation request | 1 | Brainstorm |  |  |
| SpecPtn-1f | SpecificationPattern | Can validate message creation request | 1 | Brainstorm |  |  |
| SpecPtn-2 | SpecificationPattern | Specification Pattern data retrievel from context | 1 | Brainstorm | this is done in the repository layer, so only need to test if one query work (kind of, would be best with all) |  |
| App-Com-3 | Application/Communication | Can login | 2 | Brainstorm | not in first version |  |
| App-Com-4 | Application/Communication | Can logout | 2 | Brainstorm | not in first version |  |
| App-Com-5 | Application/Communication | It should be possible to like a message on the feed page | 2 | Brainstorm | Only when logged in |  |
| Security-1 | Security | Middleware should validate if the user is logged in | 2 | Brainstorm | Only on endpoints which require login |  |
| End-1 | Endpoint | Endpoint for login | 3 | Brainstorm | not in first version |  |
| App-UI-3 | Application/UI | Login page with username and password | 4 | Brainstorm | only display when not logged in |  |
| App-UI-4 | Application/UI | Logout page | 4 | Brainstorm | only display when logged in |  |
| End-5 | Endpoint | Endpoint for registrating user | 4 | Brainstorm | not in first version |  |
| Security-5 | Security | Can create login tokens and refresh tokens | 5 | Brainstorm | JWT token |  |
| Security-4 | Security | Remove expired refresh tokens | 7 | Brainstorm | JWT token |  |
| File-1 | File Handling | Can read image files | 8 | Brainstorm | not in first version |  |
| File-2 | File Handling | Can validate file signatures | 8 | Brainstorm | not in first version |  |
| Security-3 | Security | Rename file before uploading it | 8 | Brainstorm | not in first version |  |
| File-3 | File Handling | Can upload BLOBs to storage | 9 | Brainstorm | not in first version |  |
| File-4 | File Handling | Can remove BLOBs from storage | 9 | Brainstorm | not in first version |  |
| File-5 | File Handling | Can handle file with invalid signature | 9 | Brainstorm | not in first version |  |
| CQRS-1-2 | CQRS | Backend can use commandbus | 9999 | Brainstorm | To specific and not really useful |  |
| CQRS-1-3 | CQRS | Handlers for each command | 9999 | Brainstorm | To specific and not really useful |  |
| Error-2 | Error Handling | Validation can generate a binary flag of errors | 9999 | Brainstorm | priority given it would not be tested nor really is a requirement, just an implementation design |  |
| Repo-1 | Repository | Repository for User | 9999 | Brainstorm |  |  |
| Repo-2 | Repository | Repository for Message | 9999 | Brainstorm |  |  |
| Repo-3 | Repository | Repository for Lifeform | 9999 | Brainstorm | split into 3, 1 for saving and 2 for retrieving out from specific subtypes |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Projektplanlægger** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| *Fremhæv et tidsrum til højre. Derpå følger en forklaring af diagrammet.* | | | | | **Fremhævning af tidsrum:** | 5 |  |  | Varighed for plan | | | | |  | Faktisk startdato | | | | |  | % fuldført | | |  | Faktisk (ud over plan) | | | | | |  | % fuldført (ud over plan) | | | | | | |
| **AKTIVITET** | **STARTDATO FOR PLAN** | **VARIGHED FOR PLAN** | **FAKTISK STARTDATO** | **FAKTISK VARIGHED** | **PROCENTDEL FULDFØRT** | **TIDSRUM** | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** | **29** | **30** | **31** | **32** | **33** |
| **Case Beskrivelse skrivning** | 1 | 1 | 1 | 1 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Gantt Opsætning** | 2 | 2 | 2 | 4 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Backend base struktur** | 1 | 6 | 1 | 3 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Backend implementering** | 4 | 4 | 5 | 0 | **80%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **SignalR implementering** | 0 | 2 | 0 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Produktrapport** | 1 | 23 | 1 | 0 | **20%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Processerrapport** | 6 | 15 | 4 | 0 | **10%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **App implementering** | 8 | 8 | 0 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Kravspecifikationer** | 3 | 1 | 3 | 3 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **App Test** | 14 | 6 | 0 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Backend Test** | 10 | 7 | 8 | 0 | **40%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Endpoints** | 3 | 1 | 3 | 0 | **80%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **CQRS** | 3 | 2 | 3 | 1 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Factories** | 4 | 2 | 4 | 3 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Repositories** | 3 | 1 | 3 | 1 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Result Patterns** | 1 | 1 | 1 | 1 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Security** | 14 | 4 | 0 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Services** | 4 | 3 | 4 | 0 | **80%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Specification Pattern** | 4 | 2 | 3 | 3 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **App/Communication** | 8 | 4 | 9 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **App/UI** | 8 | 3 | 9 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **App/Tech** | 8 | 6 | 9 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Context** | 2 | 1 | 2 | 3 | **100%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **App/SignalR** | 0 | 2 | 0 | 0 | **0%** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |