Sprint Grading Rubric

Topic	Unsatisfactory (0)	Developing (2-1)	Satisfactory (4-3)	Exemplary (5)
Usability	Multiple bugs during demo **OR** Features added did not improve the usefulness nor usability of the program			Few or no bugs during demo **AND** Features added made the program more useful/usable
Installation and Security	Release relies on hardcoded resource needs **OR** Installation instructions not include outside libraries that also must be installed (and this was not cleared with the professor) **OR** Private data stored on the server in an insecure manner **OR** Private data transmitted to/from the server in an insecure manner	Release is specific to screen size beyond what is appropriate for this type of program **AND** All transmissions and storage of private data takes appropriate steps to protect it	Release is specific to browser or hardware specification beyond what is appropriate for this type of program **AND** Release runs at all appropriate screen sizes (including mobile, tablet, & desktop for a web app) **AND** All transmissions and storage of private data takes appropriate steps to protect it	Release runs on all appropriate computers/browsers **AND** Release runs at all appropriate screen sizes (including mobile, tablet, & desktop for a web app) **AND** All transmissions and storage of private data takes appropriate steps to protect it
User Stories	Few, if any, user stories created or developed in sprint follow template **OR** Few, if any, features developed during sprint have an associated user story **OR** Few, if any, features added in sprint match the expectations set by their user stories and acceptance tests	Many user stories created or developed in sprint follow template **AND** Many features developed during sprint have an associated a user story **AND** Many features added in sprint match the expectations set by their user stories and acceptance tests	Most user stories created or developed in sprint follow template **AND** Most features developed during sprint have an associated a user story **AND** Most features added in sprint match the expectations set by their user stories and acceptance tests	All user stories created or developed in sprint follow template **AND** All features developed during sprint have an associated a user story **AND** All features added in sprint match the expectations set by their user stories and acceptance tests
Deliverables	Almost none or no deliverables added during sprint **OR** Cannot document work performed consistently throughout sprint	Fewer deliverables added than would be expected for duration and no other efforts documented **OR** Deliverables were added, but did not improve project viability	Useful deliverables added in sprint appropriate for duration **AND** If any deliverables were started but not completed, scrum board documents deliverable and they will be continued in the next sprint	Useful deliverables added in sprint appropriate for duration **AND** If any deliverables were started but not completed, scrum board documents deliverables, shows they will be continued in the next sprint, & are not released to customer

Topic	Unsatisfactory (0)	Developing (2-1)	Satisfactory (4-3)	Exemplary (5)
Scrum Board Organization	Scrum board does not include BACKLOG, PLANNED, IN-PROGRESS, TESTING, COMPLETED, and CLOSED pipelines **OR** 1 or more user stories or tasks developed during sprint are missing tags or incorrectly tagged **OR** 1 or more user stories or tasks are not in the correct pipeline			Scrum board includes BACKLOG, PLANNED, IN-PROGRESS, TESTING, COMPLETED and CLOSED pipelines **AND** All user stories and tasks developed during sprint are tagged correctly **AND** All user stories and tasks are in the correct pipeline
Repo Organization	Repo does not use or incorrectly uses MAIN & DEVELOP branches **OR** Repo is not organized nor documented so that an outsider can easily find where code is stored; find where tests are stored; and find where other documents are stored	Repo uses MAIN & DEVELOP branches correctly **AND** Repo is documented so that an outsider can easily find where code is stored; find where tests are stored; and find where other documents are stored	Repo uses MAIN & DEVELOP branches correctly **AND** Repo's directory names make it easy for an outsider to easily find where code is stored; find where tests are stored; and find where other documents are stored **AND** Code in the repo is organized logically	Repo uses MAIN & DEVELOP branches correctly **AND** Repo's directory names make it easy for an outsider to easily find where code is stored; find where tests are stored; and find where other documents are stored **AND** Code in the repo is organized logically
User Stories	Few or no user stories developed during sprint broken down into tasks **OR** Few or no user stories developed during sprint include acceptance test(s) *(OR Few or no branches used in sprint have names identifying the user story for which it was created)*	Many user stories developed during sprint broken down into tasks **AND** Many user stories developed during sprint include acceptance test(s) *(AND Many branches used in sprint have names identifying the user story for which it was created)*	Most user stories developed during sprint broken down into tasks **AND** Most user stories developed during sprint include acceptance test(s) *(AND Most branches used in sprint have names identifying the user story for which it was created)*	All user stories developed during sprint broken down into tasks **AND** All user stories developed during sprint include acceptance test(s) *(AND All branches used in sprint have names identifying the user story for which it was created)*

<ONBOARDING *continues on next page*>

Tasks	Few or no tasks developed during sprint added as a dependency of their user story **OR** Few or no tasks developed during sprint include task test(s)	Many tasks developed during sprint added as a dependency of their user story **AND** Many tasks developed during sprint include task test(s)	Most tasks developed during sprint added as a dependency of their user story **AND** Most tasks developed during sprint include task test(s)	All tasks developed during sprint added as a dependency of their user story **AND** All tasks developed during sprint include task test(s)
Commits Messages	Few or no commit messages clearly identify functionality being changed or why changes were needed	Many commit messages identify functionality bring changed and why commit was needed	Most commit messages identify functionality bring changed and why commit was needed **AND** Most commit messages link to the single task for which they were created	Most commit messages identify functionality bring changed and why commit was needed **AND** Most commit messages link to the single task for which they were created **AND** Most first lines provide summary of commit

Topic	Unsatisfactory (0)	Developing (2-1)	Satisfactory (4-3)	Exemplary (5)
Acceptance Test Definition	One or more user stories developed during sprint lacked acceptance test(s) **OR** One or more acceptance tests fail			Acceptance test(s) defined for every user story developed during sprint **AND** All acceptance tests pass
Acceptance Test Language	No acceptance tests can be run and approved by a typical (untrained) user **OR** Most acceptance tests include project developer-specific language/terms	Few acceptance tests can be run and approved by a typical (untrained) user **AND** Few acceptance tests include project developer-specific language/terms	Many acceptance tests can be run and approved by a typical (untrained) user **AND** No acceptance tests include project developer-specific language/terms	All or nearly all acceptance tests can be run and approved by a typical (untrained) user
Unit & Task Test Definition	User stories were not broken down into tasks **OR** Few or no tasks worked on during the sprint have task test(s) which verify the code works correctly	Most or all user stories were broken down into tasks **AND** Many tasks worked on during the sprint have task test(s) which verify the code works correctly	Most or all user stories were broken down into tasks **AND** Most tasks worked on during the sprint have task test(s) which verify the code works correctly	Most or all user stories were broken down into tasks **AND** All tasks worked on during the sprint have task test(s) which verify the code works correctly
Unit & Task Test Efficacy	Few or no tasks have task test(s) that make a convincing case the task was/will be completed correctly **OR** One or more task tests do not pass	Many tasks have task test(s) that make a convincing case the task was/will be completed correctly **AND** All task tests pass	Most tasks have task test(s) that make a convincing case the task was/will be completed correctly **AND** All task tests pass	All tasks have task test(s) that make a convincing case the task was/will be completed correctly **AND** All task tests pass