Contents

[Design Consideration 3](#_Toc487362517)

[Out of scope scenarios 3](#_Toc487362518)

[Limitations 3](#_Toc487362519)

[Technical requirements 3](#_Toc487362520)

[Functional business requirements 4](#_Toc487362521)

[Manual steps after installing package 13](#_Toc487362522)

# Design Consideration

1. As per requirement, Timesheet should have all Time entries added by different people under it. But this is not the correct way. So I have implemented to generated timesheet per user and timesheet will have time entries only for that user.
2. As per requirement, As a user, when on a project record, the time entry component already recognizes which project I’m on, I just need to start/resume the timer. But as per my understanding, it is best to put timer on task record and that’s why I have implemented in that way.
3. As per requirement, Manger should be able to see all records while user only self record. To implement this. I have kept org wide default for Project, Project Tasks, Timesheet, Time entries as ‘private’

Whenever manager user added to project, I am sharing Project and related Project Tasks, Time entries, Time sheets with this user in Edit mode. When member use is added, I am sharing Project and project tasks, and project members as ‘Read’.

In Addition to this I have created one profile and permission set to give CRUD permission to Time tracker objects.

# Out of scope scenarios

1. When User delete Project Member from Project, unshared Project, project tasks, time entries from that user.

# Limitations

1. System allows multiple project members with ‘Manager’ role on a project
2. Currently even if User is not added to project as a member, he can start and stop the timer. Ideally this should not be allowed
3. On Task record, Timer component will not be auto refreshed when you manually add time entry from related list. You need to refresh the task detail page manually.
4. When you manually add the time entry either using ‘Project Time Entry’ tab or from Project Task related list, the total time (hrs) will be calculated only after save of the record.
5. If you add manual time entry with past start and stop date. This entry will be considered as a latest and will be shown on timer component.

# Technical requirements

1. Auto-generate documentation based on commented code

**I have used following app to generated the documentation**

<https://github.com/SalesforceFoundation/ApexDoc>

1. All functionality must be in the domain layer

**I have used most of standard functionality except Timer component and sharing records with users.**

1. Use centralized trigger handling

**I have used trigger factory pattern. There is one ‘TriggerInterface’ that will be implemented by each TriggerHandler class. There is one TriggerFactory class, which will instanciate each TirggerHandler class and return TriggerInterface’ object. Trigger business logic is wrapped inside Trigger helper calss.**

1. Set up an ant build.xml for using force.com deploy tools

**Set up build.xml and pushed in github too**

1. Managed package link

[**https://login.salesforce.com/packaging/installPackage.apexp?p0=04t7F0000008cP3**](https://login.salesforce.com/packaging/installPackage.apexp?p0=04t7F0000008cP3)

1. Github link

[**https://github.com/ajoshi27/timetrackerApp**](https://github.com/ajoshi27/timetrackerApp)

# Functional business requirements

|  |  |
| --- | --- |
| Requirement | As a manager, I can manage tasks |
| Solution | User can add task navigating to Project Task Tab under Time track custom APP  APP Launcher🡪 Time Tracker App |

|  |  |
| --- | --- |
| Requirement | As a manager, I can manage people |
| Solution | 1. Manager can add new users using standard Salesforce user management. 2. Assign ‘TimeTracker Standard Profile’ profile to new user 3. Assign ‘TimeWatch Standard Access’ to new user 4. For testing purpose assign Platform licenses. |

|  |  |
| --- | --- |
| Requirement | As a manager, I can create a project |
| Solution |  |

|  |  |
| --- | --- |
| Requirement | As a manager, I can add people to the project |
| Solution |  |

|  |  |
| --- | --- |
| Requirement | As a manager, I can add tasks for a project |
| Solution |  |

|  |  |
| --- | --- |
| Requirement | As a manager, I can see and do everything a user can see and do. |
| Solution | I have built apex managed sharing   * There are two types of users can be added to Project. ‘Manager’ and ‘Member’ * If Manager is added, share current project related tasks, timesheet, time entries with ‘Edit’ access to him using triggers * If ‘member’ is added share only project and related tasks with read access |

|  |  |
| --- | --- |
| Requirement | As a user, I can log a time for a task under a project |
| Solution | User can log a time using timer component on Task page |

|  |  |
| --- | --- |
| Requirement | As a user, I can start a timer |
| Solution | User can start the time using ‘Star’ button on task page as shown in below screen |

|  |  |
| --- | --- |
| Requirement | As a user, I can pause the timer |
| Solution | User can pause the time using ‘Stop’ button as shown below |

|  |  |
| --- | --- |
| Requirement | As a user, I can resume a timer of a single time entry (I should see the latest time entry or select a previous one from that day to resume) |
| Solution | User will always see the latest (last modified) timer entry details on Timer component. If user add manual entry using Time Entry related list task with past date then that entry will be shown on time component |

|  |  |
| --- | --- |
| Requirement | As a manager, I see a weekly timesheet |
| Solution | Actually Timesheet will be created per user and not a single timesheet for all users. I think this is the right design. In real scenario, timesheet will be per user. So if there are 3 users on project then 3 timesheet will be generated. |

|  |  |
| --- | --- |
| Requirement | A new timesheet is auto-generated for each week in advance over Sunday night |
| Solution | I have created schedulable class **‘TW\_ScheduleGenerateTimesheetBatch’** and schedule it weekly using ‘Schedule apex’ option |

|  |  |
| --- | --- |
| Requirement | All time entries created during the week by people under projects also show up under that week’s timesheet record |
| Solution |  |

|  |  |
| --- | --- |
| Requirement | As a manager, I can see the aggregate $ and time amount on a timesheet |
| Solution |  |

|  |  |
| --- | --- |
| Requirement | As a manager, I can approve a weekly timesheet |
| Solution | I have created Standard Approval process for this. Once timesheet is generated, it will be sent to project manager for approval. Project manager will approve it using ‘Approve’ button under ‘Approval History’ Related list on Project Timesheet |

|  |  |
| --- | --- |
| Requirement | Once a timesheet is approved, all time entries related to it are locked |
| Solution | Time sheet will be marked as ‘Approved’ once it is approved and user can’t edit it. Only Manager and Admin can edit it. |

|  |  |
| --- | --- |
| Requirement | Not editable by anyone, unless manager unlocks the timesheet |
| Solution | Manger will uncheck ‘IsApproved’ Flag on timesheet and then user will be able to edit it |

|  |  |
| --- | --- |
| Requirement | As a user, I can’t add a new time entry to a weekly timesheet once it’s been locked by the manager |
| Solution |  |

|  |  |
| --- | --- |
| Requirement | As a user, when on a project record, the time entry component already recognizes which project I’m on, I just need to start/resume the timer |
| Solution | As per my understanding, Timer should be on Task and not on project. As user can start timer for any of the task. |

|  |  |
| --- | --- |
| Requirement | Each time entry has a $ value (based on the task, project or person’s hourly rate) |
| Solution | Navigate to ‘Timer entry’ related list on Project task where you can see the $ value entry. You can also see it on ‘Project Time Entry’ detail page |

|  |  |
| --- | --- |
| Requirement | Timer component should work on Salesforce1 mobile app |
| Solution |  |

|  |  |
| --- | --- |
| Requirement | Timer component should work on Salesforce1 mobile app |
| Solution |  |

# Manual steps after installing package

1. Create new My Domain. It is required in order to use lightning components and we have stopwatch lightning component.

2. Setup approval process

**Step: 1**

- go to setup--> process automation--> approval process

- select 'Project Timesheet' as object and click create new approval process

- Give name as 'TimeTracker - Project Timesheet Approval'

- specify criteria as project:createDate not equal blank

- put 'Next Automated Approver Determined By ' as blank

- Record editability 'Administrators OR the currently assigned approver can edit records during the approval process.'

- Approval Assignment Email Template Folder : ' Timetracker email templates' as 'Time sheet assignment notification to Manager'

- Select Available fields as you want

- Initial Submitters = 'Project Timesheet owner'

- Submission Settings = select 'Allow submitters to recall approval requests'

**Step 2**: Create new step

- Give name 'Send to Manager for Approval'

- Specify Step Criteria = 'All records should enter this step'

- Select Approver = 'Automatically assign to approver(s).'

- Select Related user = 'Timesheet Manager'

- When multiple approvers are selected: 'Approve or reject based on the FIRST response.'

**Step 3**

1. Create new Email alert and associated with above approval's Approved action

- Give name 'TimeTracker - Notify user about timesheet approval'

- Email Template = Time sheet Approval notification

- select Recipient Type = 'Owner' and Recipients='Timesheet Owner'

2. Create new Approval action as Field update

- Give name as 'Time Tracker - Mark Timesheet as approve

- select Field to Update as 'isApproved' and value equals 'true'

3. Add new Final Rejection Actions as 'Email Alert'

- Give name 'TimeTracker - Notify user about timesheet rejection'

- Email Template = Time sheet Rejection notification

- select Recipient Type = 'Owner' and Recipients='Project Timesheet Owner'

**Step 4:**

- Activate Approval process

3. Go to custom settings --> open 'Timewatch Trigger Settings' --> click on Manage and add following entries

1. Name = TimeWatch\_\_Project\_Member\_\_c Active = true

2. Name = TimeWatch\_\_Project\_Task\_\_c Active = true

3. Name = TimeWatch\_\_Project\_Timesheet\_\_c Active = true

4. Name = TimeWatch\_\_Project\_Time\_Entry\_\_c Active = true

4. Create new user and assign timetracker profile and permission set

5. Login as a new user and use the time tracker app