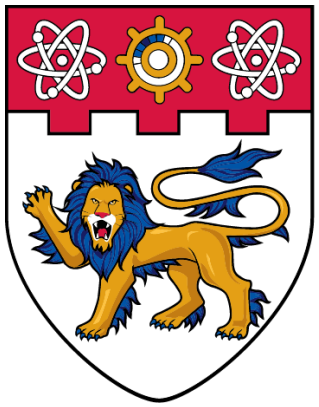

NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

School of Computer Science and Engineering

Advanced Software Engineering

Release Plan

Website Portal: HouseHunt

Supervisor: Althea Liang

13 October 2021

By Team 5:

Tang Kai Wen, Alvin

Yong Wen Shiuan

Bankata Mishra Spriha

Gupta Suhana

Heng Chor Chen, Sabrina

Heng Fuwei Esmond

Revision History

Version	Date	Primary Author(s)	Comments
0.1.0	11/10/2021	Heng Chor Chen, Sabrina Heng Fuwei Esmond	Initial Draft
1.0	20/10/2021	-	Submission of release plan

Table of Contents

Release Plan Approval

The undersigned acknowledge they have reviewed the **HouseHunt Release Plan** and agree with the approach it presents. Changes to this **Release Plan** will be coordinated with and approved by the undersigned or their designated representatives.

Signature		**Date:**	20 Oct 2021
Print Name	Tang Kai Wen, Alvin		
Title:	Project Manager		

Signature	***Heng***	**Date:**	11 Oct 2021
Print Name	Heng Chor Chen, Sabrina		
Title:	Release Manager/Engineer		

1. Introduction

The Release Plan is a tactical document that captures and tracks the features of Team HouseHunt's Web Based Application, and the upcoming release version of their date of availability.

This release plan aims to help our group determine the project's priorities and to follow the progress and developments made. The plan will also allow the team to identify criteria for the release versions based on the significance or urgency of the task, while concurrently ensuring that the project is always moving in the correct direction and that logical releases are frequently happening throughout the software development life cycle.

2. Referenced Documents

The following table summarizes the documents referenced in this document.

HouseHunt Document Name	Document Version	Issuance Date
Project Plan	1.0	01 Sep 2021
System Requirement Specification	1.0	13 Sep 2021
Quality Plan	1.0	13 Sep 2021
Project proposal	1.0	13 Sep 2021
Risk Management Plan	1.0	05 Oct 2021
Design for Software Maintainability	1.0	09 Oct 2021
Change Management Plan	1.0	14 Oct 2021
Software Configuration Management Plan	1.0	14 Oct 2021

3. Overview

The HouseHunt project has been developed for the purpose of tackling the problem in the resale market as users are often faced with confusion when searching up HDB flats. This is especially so during the Covid'19 pandemic where many Build-To-Order flats were cancelled or delayed, resulting in a surge of people going for resale flats. This application aims to simplify the vast quantity of information available online or given to them by real estate agents. The information it provides includes flat finding based on certain criteria, estimations of resale flats prices based on recent data, house affordability calculations using user's income information, and data visualisations of house prices against certain parameters such that users can understand the pricing trends in Singapore. This project has involved the development of an web-based application which only requires an internet connection and a smart device or computer. HouseHunt is built on the Django Framework for the backend server (specifically, version 3.1.7). It interfaces with the SQLite3 database using Django's in-built Object-Relational-Mapper (ORM). Django framework is extremely scalable, and the apps can be scaled reliably as web traffic increases. SQLite tends to work well with websites that generate low to medium traffic. As HouseHunt relies mostly on read access rather than write access, thus limiting its interactions with the database, there should be no issues with scalability.

Our focus on the HouseHunt application aims to be the one-stop solution to the average Singaporean's real estate needs by focusing on ease of access and reliability.

4. Assumptions, Constraints, and Risks

4.1 Assumptions

Manpower & Team Size

- There are enough people aside from the development team to help with the maintenance of the application/people are willing to take on the job of maintaining the application.
- Team of maintainers to manage the site are already trained before the release.
- There is no change in the number of team members required to complete the project.
- All the stakeholders are always present for meetings and updates.
- Developers are experienced enough to build the application.
- Developers are experienced in employing SQLite.

Time

- The time frame for the project is confirmed and finalized in the HouseHunt Project Plan.
- All tasks are completed according to the schedule.

Budget & Funding

- Funding and materials required for the project will always be on time for the tasks and fully accounted for.
- There are no additional costs added to the project.
- There are no budget cuts to the project.

Development

- Scope of the project remains the same.
- There are no significant technological modifications to be made to the development of the project.

4.2 Constraints

Manpower & Team Size

- Fixed team size of 6, results in proper assigning and planning of tasks to be done.

Time

- The project must be ready and launched before the end of the Academic Year 2021-22 Semester 1.

Budget & Funding

- Limited budget results in limitations of obtaining necessary resources needed for the project. Based on the limited budget, the project must be completed, nonetheless.
- No additional resources will be provided; therefore, resource management must be done and effectively managed.

Development

- The development of HouseHunt is limited to only an intelligent agent providing estimation of the value of real estate sellers' flats based on factors such as flat type and model, locality, remaining lease, and floor area. HouseHunt does not cater to the buying and selling of flats.

4.3 Risks

Risk Type	Risk Description	Likelihood of Occurrence	Impact	Risk Scoring Grade	Strategies to counter
Technology	Codes do not run optimally or correctly.	Moderate	**Serious** Deadline may be pushed back further	9	The development team will have to fix the error and test the solution as soon as possible.
Technology	SQLite database Breach.	Low	**Serious** Sensitive data will be exposed to the public	6	Security Groups and Network Access Control List needs to be properly configured to only allow proper inbound and outbound rules. Any possible breaches to the system database must be identified at the earliest notice and the situation must be dealt with the utmost priority. To improve the security of the system database, the server must be ideally patched several times every semester to remove the vulnerability.

Technology	SQLite cannot process the amount of traffic requests from the site.	Very Low	**Serious** Users will have problems accessing the application or it will be slower.	3	Enable auto-scaling of the number of requests to another tier such that it will automatically increase the computing amount is able to take in for HouseHunt. Any suspicious traffic (E.g., Distributed Denial-of-ServiceDDoS) must be immediately looked into by the IT security personnel.
Technology	The application is unable to process the user's request.	Low	**Serious** Users will lose trust in the website and its reliability.	6	Return proper error messages and continuously test the application for possible requests.

Technology	The application might not be sync-ing with the data.	Low	**Tolerable** Content the users query might not be different on different devices.	4	In the event that there is a change of practice in the system, ensure that it is properly communicated to the admins handling the system. A channel for admins handling the system to voice their opinion on feedback and improvements must be needed.
People	Team members have disagreements	Moderate	**Serious** Slows down progress of the project and results in bad blood between members	6	The disagreement will be brought up during team meetings to be addressed. The team leader will decide the right course of action to resolve the problem.
People	Team members fall sick	Low	**Tolerable** Causes slow down and lag in schedule.	4	Team members will be assigned to take over the sick member's role and allocated jobs respectively according to their workload.

People	Maintainers do not know how to operate the SQLite console to respond to queries.	Very Low	**Tolerable** Some issues that users have may not be immediately catered to.	2	Have a training manual to teach maintainers how to use the SQLite console and to also have at least one developer to assist if needed.
Organizational	Sponsorship/ budget cuts	Low	**Tolerable** The resources the team will be able to use might be limited.	2	Use free software as much as possible and payment may be made in instalments or delayed.
Estimation	Tasks are not completed on time	High	**Serious** The whole development will be pushed back and tasks will snowball.	12	The team members will have to prioritize the tasks that are more important and help each other out whenever they are done with their own for that period.
Requirements	There are changes to the functions and features that require major rework of the project.	Low	**Serious** The whole project will need to be changed accordingly.	6	A clear scope and SRS should be given and approved by the relevant stakeholders and agreed by all the members.

Requirements	The project gets cancelled after completion	Low	**Catastrophic** All the effort and project resources would go down to waste.	8	Resources should be reusable so that the project does not need to be duplicated or parts of it can still be used for other projects.

5. Release Approach

5.1 Rationale

HouseHunt is an application that requires a development process which needs to go through some stages of management, planning, analysis, development, and testing of codes and documentation. During *the development process, to the first release version it undergoes constant updates weekly review* on members progress *until we finally release the final version, where HouseHunt is completed and bugs free.*

A release version is important to all of us as it helps us to get HouseHunt being tested, approved, and ensures us that all versions of HouseHunt can be retrieved when we wish to do so and are not accidentally being changed by any of the development team members.

Anybody in the team is allowed to make changes to HouseHunt because every version of HouseHunt is being added with a version number which helps us differentiate the difference of the previous version and provides flexibility to backtrack changes, it can be easily reverted to the previous version if the newer version faces issues which disrupt the other functionality not to work due to the update or a major bug arise being found in the application for etc. By doing so, we are able to split the tasks into small different subcomponents therefore creation of HouseHunt will end up becoming easier to test and debug.

5.2 Release Strategy

The Release Strategy we will be using is functionality rollout. Whereby when one of the functionalities of the application is complete, it will be pushed, and a version will be created. The incremental version approach allows us to keep a monitoring of the development progress and raises any issues found in the early implementation stages if any. If any issues are to be found, the team will consider reverting back to the previous version or add a new version putting a comment bugfix based on function name (login) for etc. Proper allocation of resource management has also been provided to all members. Which allows us all to easily track the progress of the project and determine whether more resources are needed for the project.

5.3 Release Content

Based on HouseHunt SRS documentation, the system uses cases and users' requirements are being shown below:

Use Cases

1) Users must be able to **Search for a flat**
2) Users must be able to **Estimate Selling Price**
3) Users must be able to **View on Map** using google Maps API
4) Users must be able to **Calculate maximum affordability**
5) Users must be able to **View Data Visualisations**

From these different use cases, we can divide this content into their specific release version.

The development process of HouseHunt will include all the specified functionalities being mentioned above. For every release version that any members are ready to release and deploy, the information of the changes and updates are informed to all relevant personnel and project manager before being deployed.

5.4 Release Schedule

The official release of HouseHunt -Version 1.0.0 will be released in November 2021 as documented in the project schedule with all specified functionalities working perfectly and friendly for all users.

If any subsequent update releases are to be made in future, it will undergo the same process and testing before releasing a new version of HouseHunt.

5.5 Release Impacts

Every time a new version is released, It might cause some inconvenience due to application downtime causing users unable to use the application during maintenance. If the application UI or new functionality is added it might cause some unfriendliness to old users who are using the application as they are not used to it but the changes will be notified through the release notification to the users. Adding more things to HouseHunt will also cause the HouseHunt application size to be increased and hence we need to keep a lookout on memory space used to ensure that 20-30% is free as well as making sure application size is kept as minimal as possible.

5.6 Release Notification

Methods of how each user group and what information will be included in each notification prior to the release is mentioned below.

User Group	Notification Method
Team members	Emails and Meetings will be scheduled and conducted. All members will review the new codes and documentation implemented and determine whether the newer versions are better than the current version and contain all the required functionalities before releasing live for users to use.
Users	Users will be notified through an email that the application is undergoing maintenance. Once maintenance is completed, we will send another email to let users know maintenance is completed and they are able to use the application. We will also include what are the updates being done to the application

6. Acronyms

This table represents the acronyms being used in this document with their respective definitions.

Term	Definition
QA	Quality Assurance
SRS	Software Requirements Specifications