

Requirements TIAMAT:

1. Maken van ASTs

Prioriteit: Hoog

Beschrijving: Alle code die we zullen maken zal rechtstreeks verbonden zijn met een AST. Het is dus essentieel dat we ASTs kunnen maken.

Status: behaald, kan eventueel nog wat aanpassingen krijgen om alle mogelijke syntax van AT te kunnen verwerken

2. Manipuleren van ASTs

Prioriteit: Hoog

1. Maken van ASTs

Prioriteit: Hoog

Beschrijving: Alle code die we zullen maken zal rechtstreeks verbonden zijn met een AST. Het is dus essentieel dat we ASTs kunnen maken.

Status: behaald, moet indien 1 aangepast wordt ook aangepast worden.

3. Renderen van ASTs naar text

Prioriteit: Hoog

Beschrijving: De onderliggende ASTs waarmee gewerkt wordt moeten gerenderd kunnen worden naar tekst. Deze tekst wordt voorgesteld aan de gebruiker, en deze word eveneens meegegeven met de reeds bestaande officiële AmbientTalk app voor Android om geëvalueerd te worden.

Status: behaald, moet indien 1 & 2 aangepast worden ook aangepast worden

4. Colorcoding

Prioriteit: Medium

Beschrijving: Aan de hand van colorcoding van namen moet duidelijk zijn waarvoor deze staan

Status: Momenteel wordt colorcoding gebruikt om het verschil aan te tonen tussen value's (variablenamen, functienamen,..) en placeholders (rood). Dit kan nog uitgebruikt of aangepast worden om de bruikbaarheid te verhogen.

5. Pinch to zoom

Prioriteit: Medium

Beschrijving: De bestaande code moet kunnen worden in of uitgezoomd worden door pinchen zodat we bij elkaar horende code kunnen uitlichten.

Status: niet behaald

6. Evalueren code

Prioriteit: Hoog

Beschrijving: De geschreven code moet kunnen meegegeven worden aan de officiële AmbientTalk app om geëvalueerd te worden.

Status: niet behaald

6.1 Code naar textfile.

Prioriteit: Hoog

Beschrijving: De app die AT code evalueert heeft een textfile met de AT-code nodig om te evalueren. De AT code uit TIAMAT moet dus worden opgeslagen in een textfile.

Status: niet behaald.

7. Folding code

Prioriteit: Laag

Beschrijving: Zoals in Java moet samenhangende stukken code ineengefold kunnen worden. Dit gebruikt dezelfde techniek als de pinch-to-zoom, om te bepalen welke code gefold/op ingezoomd moet worden.

Status: niet behaald

8. Code correct verplaatsen.

Prioriteit: medium

Beschrijving: Verplaatsen van stukken moet tot de mogelijkheden behoren. Dit houdt in dat de scope van variabelen gecontroleerd moet worden en dat de bestaande onderliggende AST moet aangepast worden.

Status: niet behaald

9. Templates

Prioriteit: Hoog

Beschrijving: Elk programma zal beginnen van een gekozen template. Al deze templates moeten voorgedefinieerd worden, en opgelijst worden zodat de gebruiker makkelijk een template kan kiezen.

Status: Behaald, een aantal templates zijn aanwezig, voor nieuwe gedefinieerde functies worden templates aangemaakt, ook variabelen krijgen een plek tussen de templates.

Het gebruik van templates kan nog worden aangepast, zodat niet langer bij het maken van een functie een functienaam moet meegegeven worden, maar deze al direct een naam krijgt (function1 bijvoorbeeld) en dat deze naam later door de gebruiker kan aangepast worden indien gewenst. Dit zorgt ervoor dat minder typwerk essentieel is.

9.1 Templates toevoegen adhv XML files

Prioriteit: Medium

Beschrijving: Nieuwe templates toevoegen aan de hand van een XML file, ipv hardcoded in de Java code.

Status: niet behaald. Er is reeds onderzoek gebeurd om dit mogelijk te maken, zonder resultaat

10. 90°draaien commentaar

Prioriteit: Medium

Beschrijving: Elke goede code wordt voorzien van commentaar. Dus binnen deze IDE moet het ook mogelijk zijn om commentaar toe te voegen. Wanneer men een stuk tekst toevoegt moet deze door deze 90° te draaien omgezet worden van gewone tekst naar commentaar.

Status: Momenteel bestaat de mogelijkheid code uit te commentarieren aan de hand van tap and hold. Deze gesture kan nog aangepast worden naar een 90° draaien gesture.

11. Commentaar inspreken

Prioriteit: Laag

Beschrijving: Google voorziet software die spraak kan omzetten naar tekst. We willen deze mogelijkheid gebruiken om commentaar toe te voegen aan code.

Status: Momenteel kan commentaar ingesproken worden en terug afgespeeld worden. Momenteel wordt deze commentaar nog niet omgezet naar tekst.

12. Extra kader voor commentaar

Prioriteit: Laag

Beschrijving: Bij value's (functienamen of variabelenamen) kan men in een extra kader commentaar toevoegen & lezen.

Status: Deels behaald. Dit extra bestaat reeds maar wordt nog niet optimaal gebruikt. Dit kader wordt ook gebruikt voor het toevoegen van gesproken commentaar.

13. Selector voor Java klassen

Prioriteit: Laag

Beschrijving: Binnen AmbientTalk kan gebruik gemaakt worden van reeds bestaande Java klassen. We willen de mogelijkheden implementeren om bestaande Java klassen te selecteren om te gebruiken. De reeds gemaakte ASTs moeten aangepast kunnen worden. Er moeten aan nodes nieuwe nodes kunnen toegevoegd worden en bestaande nodes moeten kunnen verwijderd of aangepast worden.

Status: niet behaald.